

ELECTRE TRI 2.0a
METHODOLOGICAL GUIDE
AND
USER'S MANUAL

V. Mousseau¹, R. Slowinski², P. Zielniewicz²

¹ LAMSADE, Université Paris Dauphine, Place du Maréchal De Lattre de Tassigny, 75775 Paris cedex 16, France, tel: (+33-1) 44 05 41 84, Fax: (+33-1) 44 05 40 91, email: mousseau@lamsade.dauphine.fr.

² Institute of Computing Science, Poznan University of Technology, Piotrowo 3a, 60-965 Poznan, Poland, tel: 48-618-790790, fax: 48-618-771525, e-mail: slowinsk@sol.put.poznan.pl, Piotr.Zielniewicz@cs.put.Poznan.pl.

1. GETTING STARTED	4
1.1 HARDWARE AND SOFTWARE REQUIREMENTS	4
1.2 SETTING UP THE SYSTEM	4
1.3 HOW TO GET INFORMATION	4
2. DESCRIPTION OF THE ELECTRE TRI METHOD	5
2.1 SOME BASIC CONCEPTS IN MULTIPLE CRITERIA DECISION AID	5
2.1.1 THE SORTING “PROBLEMATIC” OR PROBLEM STATEMENT	5
2.1.2 PREFERENCE MODELLING	6
2.1.3 OPERATIONAL APPROACHES TO DECISION AIDING	8
2.2 THE ELECTRE TRI METHOD	9
2.2.1 GENERAL PRESENTATION	9
2.2.2 THE OUTRANKING RELATION IN ELECTRE TRI	10
2.2.3 THE ASSIGNMENT PROCEDURES	18
2.2.4 CONSISTENCY IN THE DEFINITION OF THE CATEGORIES	21
3. ELECTRE-TRI ASSISTANT : ASSISTANCE IN THE CONSTRUCTION OF A PREFERENCE MODEL SUPPORT FOR PARAMETER ELICITATION	22
3.1 INFERENCE OF PREFERENCE PARAMETERS	22
3.2 INTERACTIVE LEARNING PROCESS	22
4. SOFTWARE COMMANDS	25
4.1 USE OF THE SOFTWARE IN A DECISION AID PROCESS	25
4.2 GENERAL ORGANISATION	25
4.3 THE FILE MENU COMMANDS	26
4.3.1 <u>N</u> EW PROJECT	26
4.3.2 <u>O</u> PEN PROJECT	26
4.3.3 <u>C</u> LOSE PROJECT	26
4.3.4 <u>S</u> AVE PROJECT	27
4.3.5 SAVE PROJECT <u>A</u> S	27
4.3.6 <u>I</u> MPORT ALTERNATIVES FROM EXCEL	27
4.3.7 PROJECT <u>R</u> EPORT	28
4.3.8 <u>P</u> RINT PROJECT	28
4.3.9 <u>P</u> RINT SETUP	28
4.3.10 <u>E</u> XIT	28
4.4 THE EDIT MENU COMMANDS	29
4.4.1 THE EDIT DATA SET COMMAND	29
4.4.2 ELECTRE TRI ASSISTANT	31
4.5 THE <u>R</u>ESULT MENU COMMANDS	34

4.5.1	ASSIGNMENT BY <u>C</u> ATEGORY	34
4.5.2	ASSIGNMENT BY <u>A</u> LTERNATIVE	34
4.5.3	<u>C</u> OMPARISON TO PROFILES	34
4.5.4	<u>P</u> ERFORMANCES OF ALTERNATIVES	34
4.5.5	<u>D</u> EGREES OF CREDIBILITY	34
4.5.6	<u>V</u> ISUALISATION OF ALTERNATIVE	34
4.5.7	<u>S</u> TATISTICS OF ASSIGNMENT	34
4.6	THE <u>W</u>INDOW MENU COMMANDS	35
4.6.1	<u>C</u> ASCADE	35
4.6.2	<u>T</u> ILE	35
4.6.3	ARRANGE <u>I</u> CONS	35
4.6.4	<u>C</u> LOSE ALL	35
4.7	<u>H</u>ELP MENU COMMANDS	35
	<u>C</u> ONTENTS	35
4.7.2	<u>S</u> EARCH TOPIC	35
4.7.3	<u>H</u> OW TO USE HELP	35
4.7.4	<u>A</u> BOUT	35
 5.A STEP-BY-STEP EXAMPLE		 36
<hr/>		
5.1	CREATING A NEW PROJECT	36
5.2	EDITING THE DATA SET	36
5.2.1	EDITING GENERAL INFORMATION	36
5.2.2	INSERTING AND EDITING CRITERIA	38
5.2.3	INSERTING AND EDITING PROFILES	42
5.2.4	INSERTING AND EDITING ALTERNATIVES	46
5.3	SAVING THE DATA SET	49
5.4	OBTAINING RESULTS	50
5.4.1	ASSIGNMENT BY CATEGORY	50
5.4.2	ASSIGNMENT BY ALTERNATIVE	50
5.4.3	INTERMEDIARY RESULTS	50
5.6	OPENING AN EXISTING PROJECT	54
5.7	LOADING DATA FROM THE PREVIOUS VERSION OF ELECTRE TRI	55
5.8	LOADING A SET OF ALTERNATIVES FROM EXCEL	56
5.9	AN ELECTRE TRI ASSISTANT SESSION	58
7.	REFERENCES	68
<hr/>		

1. Getting started

1.1 Hardware and software requirements

The ELECTRE TRI Software version 2.0a is developed with the C++ programming language using the Microsoft Windows interface. The hardware and software requirements are the following:

- IBM-PC compatible computer with the following minimal RAM memory requirements:
 - Win 3.1: 8 Mb,
 - Win 95: 16Mb.
- Microsoft Windows 3.1, 95 or higher.

1.2 Setting up the system

In order to install the ELECTRE TRI 2.0a software on your computer, please proceed as follows:

1. Create an Electre Tri 2.0 directory on your hard disk,
2. Insert the installation disk,
3. Copy the self-extracting file `electre.exe` in the directory you have created,
4. Run the `exe` self-extracting file from your hard disk.

1.3 How to get information

ELECTRE TRI is an existing multicriteria sorting method (see [Yu 92] and [Roy & Bouyssou 93]). The ELECTRE TRI 2.0 Software has been developed through a collaboration of two research teams :

- LAMSADE Laboratory³, University of Paris-Dauphine, France
- Institute of Computing Science⁴, Poznan University of Technology, Poznan, Poland

To order the software, ask questions or make remarks, please contact :

Dominique Vallée
Lamsade - Université Paris Dauphine
Place du Maréchal De Lattre de Tassigny
75 775 Paris cedex 16 - France
tel: (33 1) 44 05 44 72
fax: (33 1) 44 05 40 91
email : `vallee@lamsade.dauphine.fr`

³ LAMSADE - Université Paris-Dauphine, Place du M^{al} De Lattre de Tassigny, 75 775 Paris cedex 16, France.

⁴ Institute of Computing Science, Poznan University of Technology, Piotrowo 3A, 60-965 Poznan, Poland.

2. Description of the ELECTRE TRI method

2.1 Some basic concepts in Multiple Criteria Decision Aid

2.1.1 The sorting "problematic" or problem statement

In a given decision situation, it is possible to formulate the problem in different terms. Three different "problematics", i.e., problem formulations (choice, sorting and ranking) may guide the analyst in structuring the problem (see [Bana e Costa, 1996], [Bana e Costa, 1992]).

Among these problematics, a major distinction concerns relative versus absolute judgement of alternatives. This distinction refers to the way alternatives are considered and to the type of result expected from the analysis.

In the first case, alternatives are directly compared one to each other and the results are expressed using the comparative notions of better and worse. Choice (Selecting a subset A^* of the best alternatives from A , see Figure 1) or ranking (definition of a preference order on A , see Figure 1) are typical examples of comparative judgements. The presence (or absence) of an alternative a_k in the set of best alternatives A^* results from the comparison of a_k to the other alternatives. Similarly, the position of an alternative in the preference order depends on its comparison to the others.

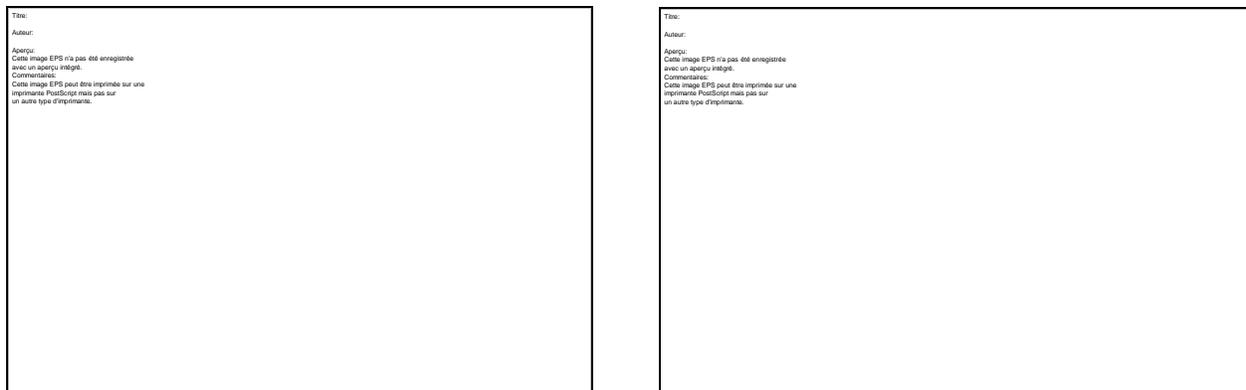


Figure 1: Choice and ranking "problematics"

In the second case, each alternative is considered independently from the others in order to determine its intrinsic value by means of comparisons to norms or references; results are expressed using the absolute notions of "assign" or "not assign" to a category, "similar" or "not similar" to a reference profile, "adequate" or "not adequate" to some norms. The sorting problematic (see Figure 2) refers to absolute judgement. It consists of assigning each alternative to one of the pre-existing categories which are defined by norms or typical elements of the category. The assignment of an alternative a_k results from the intrinsic evaluation of a_k on criteria and from the norms defining the categories (the assignment of a_k to a specific category does not influence the category to which another alternative should be assigned).

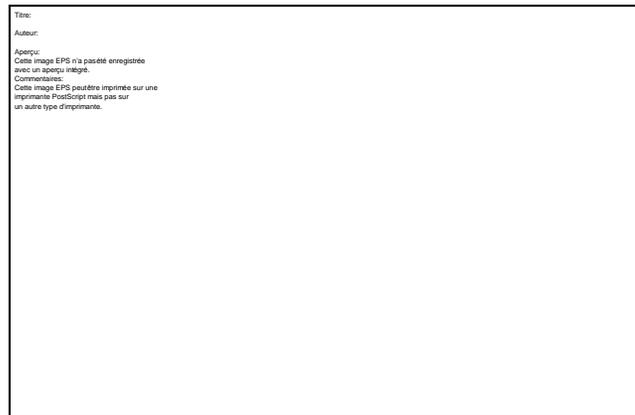


Figure 2: Sorting "problematic"

The semantic of the categories can imply an ordered structure on categories or not; the former case refers to ordered Multiple Criteria Sorting Problems (MCSP), the latter to nominal MCSP. MCSP differs from standard classification approach; the categories considered here are defined a priori and do not result from the analysis. These categories are usually conceived in such a way that alternatives assigned to the same category should be treated identically.

Previous works on MCSP have been developed using the outranking approach and several methods have been proposed: Trichotomic Segmentation (see [Moscarolla & Roy 77] and [Roy 81]), N-Tomic (see [Massaglia & Ostanello 91]), ORClass ([Larichev *et al.* 86] and [Larichev & Moshkovich 94]), ELECTRE TRI (see [Yu 92a], [Yu 92b] and [Roy & Bouyssou 93]). Filtering methods based on concordance and non-discordance principles have been studied in [Perny1aor98]. The use of rough sets theory ([Pawlak & Slowinski 94], [Slowinski 92] and [Greco *et al.* 98a]) has also allowed significant progress in this field.

Real world case studies of MCSP have been reported in the literature in various domains:

- evaluation of applicants for loans or grants ([Groleau *et al.* 95], [Veilleux *et al.* 96] and [Greco *et al.* 98b]),
- business failure risk assessment ([Dimitras *et al.* 95], [Andenmatten 95]),
- screening methods prior to project selection ([Anandalingam & Olsson 89]),
- satellite shot planning ([Gabrel 94]),
- medical diagnosis ([Slowinski 92], [Tanaka *et al.* 92]),.

2.1.2 Preference modelling

Almost all decision aid studies involve the comparison of alternatives (either relative comparisons of couples of alternatives from A or comparisons of alternatives to norms represented by fictitious alternatives). The comparison of alternatives is naturally grounded on the consequences and attributes of these alternatives. We call criterion a real-valued function g that takes into account a specific viewpoint (grouping a class of homogenous consequences).

More precisely, a criterion g is a real-valued function mapping from A to \mathfrak{R} , such that the comparison of any pair of alternatives a and b may be grounded on the comparison of the two values $g(a)$ and $g(b)$.

The direction of preference on a criterion can be increasing or decreasing. In the first case, the higher the evaluation $g(a)$, the better is a with respect to the criterion g (quality criterion); in the second case, a criterion with a decreasing direction of preference corresponds to a criterion g on which the performance of an alternative a decreases when $g(a)$ increases (cost criterion). Without any loss of generality, we will suppose in this section that the direction of preference is increasing for all criteria.

Ideally, so as to enable the comparison of any pair of alternatives a and b in A , a criterion g should be constructed such that :

$$\begin{cases} g(a) = g(b) \Rightarrow aI_g b \\ g(a) > g(b) \Rightarrow aP_g b \end{cases} \quad [1]$$

where I_g and P_g denote the indifference and strict preference relations relatively to criterion g .

In practical situations, the evaluation of alternatives are very often subject to imprecision, uncertainty and ill determination. Consequently, a small difference of evaluation $g(a)-g(b)$ can also imply an indifference situation. Moreover, even when this difference does not seems negligible, it does not always reflect a preference situation.

It is then more reasonable and prudent to consider a more general model of criterion (the preceding one being a specific case) in which the function g should be constructed so that: $g(a) \geq g(b) \Rightarrow aS_g b$ where $aS_g b$ means “ a is at least as good as b ” (or a outranks b) according to criterion g .

In order to account for the imprecision, uncertainty and ill determination of the data, it is common to use discrimination thresholds that identify the limits between situations of indifference and strict preference. Two values q and p are introduced such that :

$$\begin{cases} |g(a) - g(b)| \leq q \Rightarrow aI_g b \\ q < g(a) - g(b) \leq p \Rightarrow aQ_g b \\ g(a) - g(b) > p \Rightarrow aP_g b \end{cases} \quad [2]$$

where Q_g denotes the weak preference relation relatively to criterion g . A weak preference relation is an intermediate situation that account for a hesitation between the situations of indifference and strict preference.

q and p are called indifference and preference threshold, respectively. In the general case, these thresholds may vary with the evaluations (see Figure 3).

The model of true-criterion defined in [1] corresponds to the case where $p=q=0$. The general model ($q \geq 0$ and $p \geq 0$) is called pseudo-criterion ; two interesting specific cases are the semi-criterion when $q=p$ and the pre-criterion when $q=0$.

Assigning a value to these thresholds is a difficult practical problem. Such values can be either determined after analysing the imprecision of the data or inferred using the ELECTRE TRI Assistant functionalities (see section 3). However, it is important to stress that, in a constructivist approach, it seems illusory to try to approximate “true value” for these parameters. These thresholds should take “reasonable values” whose impact are to be studied through a

robustness analysis. It consists in exploring the impact of the variations of the parameters on the strength of the resulting conclusions.

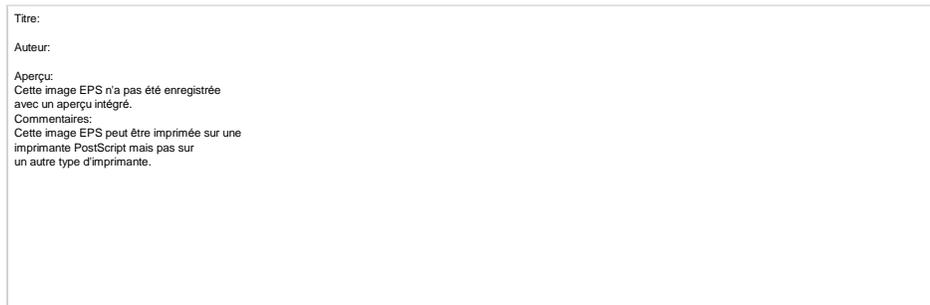


Figure 3: Pseudo-criterion

2.1.3 Operational approaches to decision aiding

In multicriteria analysis, there are three ways to proceed when facing a problem for which a prescription or recommendation is to be derived through the aggregation of performances of alternatives.

2.1.3.1 Use of a single synthesising criterion

When the criteria are rather homogenous, and total compensation between criteria is acceptable, it is frequent to build a single criterion function that accounts for all pertinent aspects of the problem. In this case, the evaluations of an alternative may be synthesised in a single value. Alternatives are then mutually comparable, as the comparisons is made by the mean of comparison s of numbers. Moreover, this way to proceed induce a transitive preference relation. Let us remark that classical methods such as the weighted sum typically refers to this approach.

2.1.3.2 Synthesising by outranking with incomparabilities

This approach relies on an aggregation rule that allows situations of incomparability, refusing a priori a total compensation between criteria. Incomparability is accepted so as to avoid arbitrary or fragile judgements. Moreover, transitivity of the outranking relation is not systematically imposed. Let us note that all ELECTRE type methods refer to this approach.

2.1.3.3 Interactive local judgements with trial-and-error iterations

This approach consist in highlighting a prescription for the decision problem through a sequence of question-answer. Each interactive procedure is grounded on an interaction protocol composed of dialogue and computing phases. The interaction stops when the decision maker is satisfied with the last proposal of the procedure.

2.2 The ELECTRE TRI method

2.2.1 General presentation

ELECTRE TRI assigns alternatives to predefined categories. The assignment of an alternative a results from the comparison of a with the profiles defining the limits of the categories. Let F denote the set of the indices of the criteria g_1, g_2, \dots, g_m ($F=\{1, 2, \dots, m\}$) and B the set of indices of the profiles defining $p+1$ categories ($B=\{1,2,\dots,p\}$), b_h being the upper limit of category C_h and the lower limit of category C_{h+1} , $h=1, 2, \dots,p$ (see Figure 4). In what follows, we will assume, without any loss of generality, that preferences increase with the value on each criterion.

Schematically, Electre Tri assigns alternatives to categories following two consecutive steps :

- construction of an outranking relation S that characterises how alternatives compare to the limits of categories,
- exploitation (through assignment procedures) of the relation S in order to assign each alternative to a specific category.

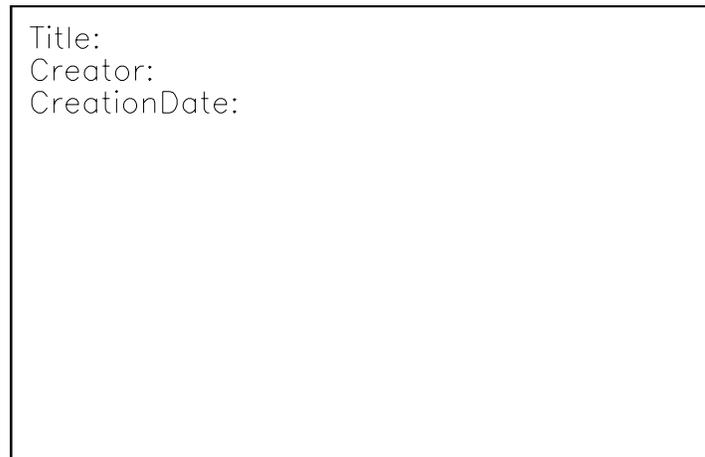


Figure 4: Ordered categories defined by limit profiles

ELECTRE TRI builds an outranking relation S , i.e., validates or invalidates the assertion aSb_h (and b_hSa), whose meaning is " a is at least as good as b_h ". Preferences restricted to the significance axis of each criterion are defined through pseudo-criteria (see [Roy & Vincke 84] for details on this double-threshold preference representation). The indifference and preference thresholds ($q_j(b_h)$ and $p_j(b_h)$) constitute the intra-criterion preferential information. They account for the imprecise nature of the evaluations $g_j(a)$ (see [Roy 89]). $q_j(b_h)$ specifies the largest difference $g_j(a)-g_j(b_h)$ that preserves indifference between a and b_h on criterion g_j ; $p_j(b_h)$ represents the smallest difference $g_j(a)-g_j(b_h)$ compatible with a preference in favor of a on criterion g_j .

At the comprehensive level of preferences, in order to validate the assertion aSb_h (or b_hSa), two conditions should be verified:

- concordance: for an outranking aSb_h (or b_hSa) to be accepted, a "sufficient" majority of criteria should be in favour of this assertion,
- non-discordance: when the concordance condition holds, none of the criteria in the minority should oppose to the assertion aSb_h (or b_hSa) in a "too strong way".

Two types of inter-criteria preference parameters intervene in the construction of S:

- the set of weight-importance coefficients (k_1, k_2, \dots, k_m) is used in the concordance test when computing the relative importance of the coalitions of criteria being in favour of the assertion aSb_h ,
- the set of veto thresholds $(v_1(b_h), v_2(b_h), \dots, v_m(b_h))$, " $h\hat{I}B$ ", 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 .

2.2.2 The outranking relation in ELECTRE TRI

In the ELECTRE TRI method, an outranking relation is build in order to enable the comparison of an alternative a to a profile b_h . This outranking relation is build through the following steps:

- compute the partial concordance indices $c_j(a, b_h)$ and $c_j(b_h, a)$,
- compute the overall concordance indices $c(a, b_h)$,
- compute the partial discordance indices $d_j(a, b_h)$ and $d_j(b_h, a)$,
- compute the fuzzy outranking relation grounded on the credibility indices $s(a, b_h)$,
- determine a λ -cut of the fuzzy relation in order to obtain a crisp outranking relation.

2.2.2.1 Partial concordance indices

The partial concordance index $c_j(a, b_h)$ ($c_j(b_h, a)$, respectively) expresses to which extent the statement " a is at least as good as b_h (b_h is at least as good as a , respectively) considering criterion g_j ". When g_j has an increasing direction of preference, index $c_j(a, b_h)$ and $c_j(b_h, a)$ are computed as follows (see Figure 5):

$$\left. \begin{array}{l} \text{if } g_j(a) \leq g_j(b_h) - p_j(b_h), \text{ then } c_j(a, b_h) = 0 \\ \text{if } g_j(b_h) - p_j(b_h) < g_j(a) \leq g_j(b_h) - q_j(b_h), \\ \text{then } c_j(a, b_h) = \frac{[g_j(a) - g_j(b_h) + p_j(b_h)]}{[p_j(b_h) - q_j(b_h)]} \\ \text{if } g_j(b_h) - q_j(b_h) < g_j(a), \text{ then } c_j(a, b_h) = 1 \end{array} \right\} [3]$$

$$\left. \begin{array}{l}
\text{if } g_j(a) \geq g_j(b_h) + p_j(b_h), \text{ then } c_j(b_h, a) = 0 \\
\text{if } g_j(b_h) + q_j(b_h) \leq g_j(a) < g_j(b_h) + p_j(b_h), \\
\text{then } c_j(b_h, a) = \frac{[g_j(b_h) - g_j(a) + p_j(b_h)]}{[p_j(b_h) - q_j(b_h)]} \\
\text{if } g_j(a) < g_j(b_h) + q_j(b_h), \text{ then } c_j(b_h, a) = 1
\end{array} \right\} [4]$$

Title: Creator: CreationDate:	Title: Creator: CreationDate:
-------------------------------------	-------------------------------------

Figure 5: $c_j(b_h, a)$ and $c_j(a, b_h)$, increasing direction of preference

When g_j has a decreasing direction of preference, index $c_j(a, b_h)$ and $c_j(b_h, a)$ are computed as follows (see Figure 6):

$$\left. \begin{array}{l}
\text{if } g_j(a) \geq g_j(b_h) + p_j(b_h), \text{ then } c_j(a, b_h) = 0 \\
\text{if } g_j(b_h) + q_j(b_h) \leq g_j(a) \leq g_j(b_h) + p_j(b_h), \\
\text{then } c_j(a, b_h) = \frac{[g_j(b_h) - g_j(a) + p_j(b_h)]}{[p_j(b_h) - q_j(b_h)]} \\
\text{if } g_j(b_h) + q_j(b_h) > g_j(a), \text{ then } c_j(a, b_h) = 1
\end{array} \right\} [5]$$

$$\left. \begin{array}{l}
\text{if } g_j(a) \leq g_j(b_h) - p_j(b_h), \text{ then } c_j(b_h, a) = 0 \\
\text{if } g_j(b_h) - p_j(b_h) \leq g_j(a) < g_j(b_h) - q_j(b_h), \\
\text{then } c_j(b_h, a) = \frac{[g_j(a) - g_j(b_h) + p_j(b_h)]}{[p_j(b_h) - q_j(b_h)]} \\
\text{if } g_j(a) > g_j(b_h) - q_j(b_h), \text{ then } c_j(b_h, a) = 1
\end{array} \right\} [6]$$

Title: Creator: CreationDate:	Title: Creator: CreationDate:
-------------------------------------	-------------------------------------

Figure 6: $c_j(b_h, a)$ and $c_j(a, b_h)$, decreasing direction of preference

2.2.2.2 Global concordance indices

Global concordance indices $c(b_h, a)$ ($c(a, b_h)$, respectively) express to which extend the evaluations of a and b_h on all criteria are concordant with the assertion “ a outranks b_h ” (“ b_h outranks a ”, respectively). $c(b_h, a)$ and $c(a, b_h)$ are computed as follows:

$$\begin{aligned}
 c(a, b_h) &= \frac{\sum_{j \in F} k_j c_j(a, b_h)}{\sum_{j \in F} k_j} \\
 c(b_h, a) &= \frac{\sum_{j \in F} k_j c_j(b_h, a)}{\sum_{j \in F} k_j}
 \end{aligned}
 \tag{7}$$

2.2.2.3 Discordance indices

The partial discordance index $d_j(a, b_h)$ ($d_j(b_h, a)$, respectively) expresses to which extend the criterion g_j is opposed to the assertion “ a is at least as good as b_h ”, i.e., “ a outranks b_h ” (“ b_h is at least as good as a ”, respectively). A criterion g_j is said to be discordant with to assertion “ a outranks b_h ” is on this criterion b_h is preferred to a ($b_h P a$, i.e., $c_j(b_h, a)=1$ and $c_j(a, b_h)=0$). In the case of increasing preferences, the criterion g_j opposes a veto when the difference $g_j(b_h)-g_j(a)$ exceeds the veto threshold $v_j(b_h)$.

When g_j has an increasing direction of preference, $d_j(a, b_h)$ and $d_j(b_h, a)$ are computed as follows (see Figure 7):

$$\left. \begin{aligned}
& \text{if } g_j(a) > g_j(b_h) - p_j(b_h), \text{ then } d_j(a, b_h) = 0 \\
& \text{if } g_j(b_h) - v_j(b_h) < g_j(a) \leq g_j(b_h) - p_j(b_h), \\
& \text{then } d_j(a, b_h) = \frac{[g_j(b_h) - g_j(a) - p_j(b_h)]}{[v_j(b_h) - p_j(b_h)]} \\
& \text{if } g_j(b_h) - v_j(b_h) \geq g_j(a), \text{ then } d_j(a, b_h) = 1
\end{aligned} \right\} [8]$$

$$\left. \begin{aligned}
& \text{if } g_j(a) \leq g_j(b_h) + p_j(b_h), \text{ then } d_j(b_h, a) = 0 \\
& \text{if } g_j(b_h) + p_j(b_h) < g_j(a) \leq g_j(b_h) + v_j(b_h), \\
& \text{then } d_j(b_h, a) = \frac{[g_j(a) - g_j(b_h) - p_j(b_h)]}{[v_j(b_h) - p_j(b_h)]} \\
& \text{if } g_j(a) > g_j(b_h) + v_j(b_h), \text{ then } d_j(b_h, a) = 1
\end{aligned} \right\} [9]$$

Title:
Creator:
CreationDate:

Title:
Creator:
CreationDate:

Figure 7: $d_j(a, b_h)$ and $d_j(b_h, a)$, increasing direction of preference

When g_j has a decreasing direction of preference, $d_j(a, b_h)$ and $d_j(b_h, a)$ are computed as follows (see Figure 8):

$$\left. \begin{aligned}
& \text{if } g_j(a) \leq g_j(b_h) + p_j(b_h), \text{ then } d_j(a, b_h) = 0 \\
& \text{if } g_j(b_h) + p_j(b_h) < g_j(a) \leq g_j(b_h) + v_j(b_h), \\
& \text{then } d_j(a, b_h) = \frac{[g_j(a) - g_j(b_h) - p_j(b_h)]}{[v_j(b_h) - p_j(b_h)]} \\
& \text{if } g_j(b_h) + v_j(b_h) < g_j(a), \text{ then } d_j(a, b_h) = 1
\end{aligned} \right\} [10]$$

$$\left. \begin{array}{l}
\text{if } g_j(a) > g_j(b_h) - p_j(b_h), \text{ then } d_j(b_h, a) = 0 \\
\text{if } g_j(b_h) - v_j(b_h) < g_j(a) \leq g_j(b_h) - p_j(b_h), \\
\text{then } d_j(b_h, a) = \frac{[g_j(b_h) - g_j(a) - p_j(b_h)]}{[v_j(b_h) - p_j(b_h)]} \\
\text{if } g_j(a) \leq g_j(b_h) - v_j(b_h), \text{ then } d_j(b_h, a) = 1
\end{array} \right\} [11]$$

Title:
Creator:
CreationDate:

Title:
Creator:
CreationDate:

Figure 8: $d_j(a, b_h)$ and $d_j(b_h, a)$, decreasing direction of preference

2.2.2.4 Degree of credibility of the outranking relation

The degree of credibility of the outranking relation $s(a, b_h)$ ($s(b_h, a)$, respectively) expresses to which extent “a outranks b_h ” (“ b_h outranks a”, respectively) according to the global concordance index $c(a, b_h)$ and to the discordance indices $d_j(a, b_h)$, $\forall j \in F$ (according to the global concordance index $c(b_h, a)$ and to the discordance indices $d_j(b_h, a)$), $\forall j \in F$, respectively). Computing the credibility indices $s(a, b_h)$ and $s(b_h, a)$ amounts at establishing a valued outranking relation.

The computation of the credibility index $s(a, b_h)$ is grounded on the following principles:

- 1) when no criteria are discordant, the credibility of the outranking relation $s(a, b_h)$ is equal to the concordance index $s(a, b_h)$,
- 2) when a discordant criterion opposes a veto to the assertion “a outranks b_h ” (i.e., $d_j(a, b_h) = 1$), then credibility index $s(a, b_h)$ becomes null (the assertion “a outranks b_h ” is not credible at all),
- 3) when a discordant criterion is such that $c(a, b_h) < d_j(a, b_h) < 1$, the credibility index $s(a, b_h)$ becomes lower than the concordance index $c(a, b_h)$, due to the effect of the opposition on this criterion.

It results from these principles that the credibility index $s(a, b_h)$ corresponds to the concordance index $c(a, b_h)$ weakened by eventual veto effects. More precisely, the value of $s(a, b_h)$ is computed as follows ($s(b_h, a)$ is computed similarly):

$$s(a, b_h) = c(a, b_h) \prod_{j \in \bar{F}} \frac{1 - d_j(a, b_h)}{1 - c(a, b_h)} \text{ where } \bar{F} = \{j \in F / d_j(a, b_h) > c(a, b_h)\} \quad [12]$$

2.2.2.5 Resulting outranking relation

The translation of the obtained fuzzy outranking relation into a crisp outranking relation S is done by means of a I -cut, (I is called cutting level). I is considered as the smallest value of the credibility index compatible with the assertion “ a outranks b_h ”, i.e., $s(a, b_h) \geq I \Rightarrow a S b_h$.

We define the binary relations f (preference), I (indifference) and R (incomparability) as follow (see Figure 9):

- $a I b_h \hat{U} a S b_h \text{ and } b_h S a$
- $a f b_h \hat{U} a S b_h \text{ and not } b_h S a$
- $a p b_h \hat{U} \text{ not } a S b_h \text{ and } b_h S a$
- $a R b_h \hat{U} \text{ not } a S b_h \text{ and not } b_h S a$



Figure 9: Definition of the binary relations f , I and R

2.2.2.6 A numerical example

Let us consider three alternatives a_1 , a_2 and a_3 evaluated on five criteria g_1 , g_2 , g_3 , g_4 and g_5 . Let us suppose that the direction of preference on each criterion is increasing and that maximum and minimum evaluation on all criteria is 100 and 0, respectively. The evaluation matrix is given in Table 1.

	g_1	g_2	g_3	g_4	g_5
a_1	75	67	85	82	90
a_2	28	35	70	90	95
a_3	45	60	55	68	60

Table 1: Evaluation matrix

Let us suppose that the alternatives are to be compared to the profile $b=(70, 75, 80, 75, 85)$ using the preferential information given in Table 2:

	g_1	g_2	g_3	g_4	g_5
k_j	1	1	1	1	1
$q_j(b)$	5	5	5	5	5
$p_j(b)$	10	10	10	10	10
$v_j(b)$	30	30	30	30	30

Table 2: Preference parameters

Comparison of a_1 and b :

- 1) computation of partial concordance indices $c_j(b, a_1)$ and $c_j(a_1, b)$, (see §2.2.2.1)

	g_1	g_2	g_3	g_4	g_5
$c_j(a_1, b)$	1	0.4	1	1	1
$c_j(b, a_1)$	1	1	1	1	1

Table 3: Partial concordance indices $c_j(b, a_1)$ and $c_j(a_1, b)$

- 2) computation of the concordance indices $c(b, a_1)$ and $c(a_1, b)$,
 $c(a_1, b) = \frac{1}{5} \cdot [(1 \times 1) + (1 \times 0.4) + (1 \times 1) + (1 \times 1) + (1 \times 1)] = 0.88$
 $c(b, a_1) = \frac{1}{5} \cdot [(1 \times 1) + (1 \times 1) + (1 \times 1) + (1 \times 1) + (1 \times 1)] = 1$

- 3) computation of the discordance indices $d_j(a_1, b)$ and $d_j(b, a_1)$,

	g_1	g_2	g_3	g_4	g_5
$d_j(a_1, b)$	0	0	0	0	0
$d_j(b, a_1)$	0	0	0	0	0

Table 4: Discordance indices $d_j(b, a_1)$ and $d_j(a_1, b)$

- 4) computation of the credibility indices $s(a_1, b)$ and $s(b, a_1)$,
 $s(a_1, b) = c(a_1, b) = 0.88$
 $s(b, a_1) = c(b, a_1) = 1$ as $d_j(a_1, b) = d_j(b, a_1) = 0, \forall j \in F$

- 5) determination of the preference relation between a_1 and b ($\lambda=0.75$)

$$\left. \begin{array}{l} s(a_1, b) \geq I \Rightarrow a_1 S b \\ s(b, a_1) \geq I \Rightarrow b S a_1 \end{array} \right\} \Rightarrow a_1 I b$$

Comparison of a_2 and b :

- 1) computation of the partial concordance indices $c_j(b, a_2)$ and $c_j(a_2, b)$, (see §2.2.2.1)

	g_1	g_2	g_3	g_4	g_5
$c_j(a_2, b)$	0	0	0	1	1
$c_j(b, a_2)$	1	1	1	0	0

Table 5: Partial concordance indices $c_j(b, a_2)$ and $c_j(a_2, b)$

- 2) computation of the concordance indices $c(b, a_2)$ and $c(a_2, b)$,
 $c(a_2, b) = \frac{1}{5} \cdot [(1 \times 0) + (1 \times 0) + (1 \times 0) + (1 \times 1) + (1 \times 1)] = 0.4$
 $c(b, a_2) = \frac{1}{5} \cdot [(1 \times 1) + (1 \times 1) + (1 \times 1) + (1 \times 0) + (1 \times 0)] = 0.6$

- 3) computation of the discordance indices $d_j(a_2, b)$ and $d_j(b, a_2)$,

	g_1	g_2	g_3	g_4	g_5
$d_j(a_2, b)$	1	1	0	0	0
$d_j(b, a_2)$	0	0	0	0.25	0

Table 6: Discordance indices $d_j(b, a_2)$ and $d_j(a_2, b)$

- 4) computation of the credibility indices $s(a_2, b)$ and $s(b, a_2)$,
 $s(a_2, b) = 0$ as $d_j(a_2, b) = 1$, for $j = 1, 2$
 $s(b, a_2) = 0.6$ as $d_j(b, a_2) < c(a_2, b)$, $\forall j \in F$

- 5) determination of the preference relation between a_2 and b ($\lambda = 0.75$)
 $\left. \begin{array}{l} s(a_2, b) < I \Rightarrow \text{not } a_2 S b \\ s(b, a_2) < I \Rightarrow \text{not } b S a_2 \end{array} \right\} \Rightarrow a_2 R b$

Comparison of a_3 and b :

- 1) computation of the partial concordance indices $c_j(b, a_3)$ and $c_j(a_3, b)$, (see §2.2.2.1)

	g_1	g_2	g_3	g_4	g_5
$c_j(a_3, b)$	0	0	0	0.6	1
$c_j(b, a_3)$	1	1	1	1	1

Table 7: Partial concordance indices $c_j(b, a_3)$ and $c_j(a_3, b)$

2) computation of the concordance indices $c(b, a_3)$ and $c(a_3, b)$,

$$c(a_3, b) = \frac{1}{5} \cdot [(1 \times 0) + (1 \times 0) + (1 \times 0) + (1 \times 0.6) + (1 \times 0)] = 0.12$$

$$c(b, a_3) = \frac{1}{5} \cdot [(1 \times 1) + (1 \times 1) + (1 \times 1) + (1 \times 1) + (1 \times 1)] = 1$$

3) computation of the discordance indices $d_j(a_3, b)$ and $d_j(b, a_3)$,

	g_1	g_2	g_3	g_4	g_5
$d_j(a_3, b)$	0.75	0.25	0.75	0	0.75
$d_j(b, a_3)$	0	0	0	0	0

Table 8: Discordance indices $d_j(b, a_3)$ and $d_j(a_3, b)$

4) computation of the credibility indices $s(a_3, b)$ and $s(b, a_3)$,

$$s(a_3, b) = 0.12 \times \frac{1-0.75}{1-0.12} \times \frac{1-0.25}{1-0.12} \times \frac{1-0.75}{1-0.12} \times \frac{1-0.75}{1-0.12} \approx 0$$

$$s(b, a_3) = c(b, a_3) = 1 \text{ as } d_j(b, a_3) = 0, \forall j \in F$$

5) determination of the preference relation between a_3 and b_h ($\lambda=0.75$)

$$\left. \begin{array}{l} s(a_3, b) < I \Rightarrow \text{not } a_3 S b \\ s(b, a_3) \geq I \Rightarrow b S a_3 \end{array} \right\} \Rightarrow a_3 \mathbf{p} b$$

2.2.3 The assignment procedures

The role of the exploitation procedure is then 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. Two assignment procedures are available.

2.2.3.1 the pessimistic assignment procedure

Pessimistic (or conjunctive) procedure :

- compare a successively to b_i , for $i=p, p-1, \dots, 0$,
- b_h being the first profile such that $a S b_h$, assign a to category C_{h+1} ($a \text{ @ } C_{h+1}$).

If b_{h-1} and b_h denote the lower and upper profile of the category C_h , the pessimistic procedure assigns alternative a to the highest category C_h such that a outranks b_{h-1} , i.e., $a S b_{h-1}$. When using this procedure with $I=1$, an alternative a can be assigned to category C_h only if $g_j(a)$ equals or exceeds $g_j(b_{h-1})$ (by some threshold) for each criterion (conjunctive rule). When I decreases, the conjunctive characters of this rule is weakened.

2.2.3.2 the optimistic assignment procedure

Optimistic (or disjunctive) procedure :

- compare a successively to b_i , $i=1, 2, \dots, p$,

b) b_h being the first profile such that $b_h \succ a$, assign a to category C_h ($a \rightarrow C_h$).

The optimistic (or disjunctive) procedure assigns a to the lowest category C_h for which the upper profile b_h is preferred to a , i.e., $b_h \succ a$. When using this procedure with $I=1$, an alternative a can be assigned to category C_h when $g_j(b_h)$ exceeds $g_j(a)$ (by some threshold) at least for one criterion (disjunctive rule). When I decreases, the disjunctive character of this rule is weakened.

2.2.3.3 Comparison of the two assignment procedures

The ideas that ground the two assignment procedures are different; consequently, it is not surprising that these assignment procedures might assign some alternatives to different categories. The following result explains, on a theoretical level the reason of potential divergence of the assignment results.

Let us suppose that an alternative a is assigned to C_i and C_j by the pessimistic and optimistic assignment rule respectively. It holds :

- C_i is lower or equal to C_j ($i \leq j$),
- C_i is greater than C_j when a is incomparable with all profiles between C_i and C_j ($a R b_i, \exists f$ such that $j \neq f < i$).

More specifically :

- when the evaluation of an alternative are between the two profiles of a category on each criterion, then both procedure assign this alternative to this category,
- a divergence exists among the results of the two assignment procedures only when an alternative is incomparable to one or several profiles; in such case the pessimistic assignment rule assigns the alternative to a lower category than the optimistic one.

2.2.3.4 Illustrative example

Let us consider the example presented in §2.2.2.6 in which we face a trichotomic segmentation problem. The three categories C_1 , C_2 and C_3 are delimited by two profiles b_1 and b_2 . b_1 represents the "frontier" between C_1 and C_2 (b_1 is the lower limit of C_2 and the upper limit of C_1) b_2 represents the "frontier" between C_2 and C_3 (b_2 is the lower limit of C_3 and the upper limit of C_2). These two profiles are defined in Table 9. b_0 and b_3 are two extreme profiles representing the anti-ideal and ideal alternatives (it holds $a \succ b_0$ and $b_3 \succ a$).

	g_1	g_2	g_3	g_4	g_5
b_1	50	48	55	55	60
b_2	70	75	80	75	85

Table 9: Definition of the profiles

The credibility indices of the outranking relation between the alternatives to be assigned (a_1 , a_2 and a_3) and the profiles (b_1 and b_2) is computed as defined in §2.2.2. The results are presented in Table 10.

	$s(a_i, b_1)$	$s(b_1, a_i)$	$s(a_i, b_2)$	$s(b_2, a_i)$
a_1	1	0	0.88	1
a_2	0.6	0	0	0.6
a_3	1	0.6	0	1

Table 10: Credibility indices $s(a_i, b_h)$

If we set $\lambda=0.75$, the resulting preference relations between a_i and b_h are the following (see Table 11):

	b_1	b_2
a_1	f	I
a_2	R	R
a_3	f	p

Table 11: Preference relations between a_i and b_h

Results of ELECTRE TRI pessimistic assignment procedure:

a_1 is assigned to C_3 because a_1Sb_3 does not hold but a_1Sb_2 holds.
 a_2 is assigned to C_1 because a_2Sb_3 , a_2Sb_2 and a_2Sb_1 do not hold but a_2Sb_0 holds.
 a_3 is assigned to C_2 because a_3Sb_3 and a_3Sb_2 do not hold but a_3Sb_1 holds.

Results with ELECTRE TRI optimistic assignment procedure:

a_1 is assigned to C_3 because b_0fa_1 , b_1fa_1 and b_2fa_1 do not hold but b_3fa_1 , holds.
 a_2 is assigned to C_3 because b_0fa_2 , b_1fa_2 and b_2fa_2 do not hold but b_3fa_2 , holds.
 a_3 is assigned to C_2 because b_0fa_3 and b_1fa_3 do not hold but b_2fa_3 , holds.

Let us remark that a_2 is assigned to C_3 by the optimistic assignment procedure, and to C_1 by the pessimistic assignment procedure. This difference among the two results stems from the fact that a_2 is incomparable to both profiles b_1 and b_2 .

2.2.4 Consistency in the definition of the categories

The ordered $p+1$ categories C_1, C_2, \dots, C_{p+1} are defined in ELECTRE TRI by p profiles $b_1, b_2, \dots, b_p, b_h$ being the upper limit of category C_h and the lower limit of category C_{h+1} , $h=1, 2, \dots, p$. For the categories to be consistently defined, the profiles should respect the two following conditions:

Condition 1: " $j\hat{I} F$, " $h=1..p-1, g_j(b_{h+1}) \geq g_j(b_h)$ "

This condition simply states that the categories should be ordered. As ELECTRE TRI considers ordered categories, it is not possible to use the method if this condition is not fulfilled.

Condition 2: " $j\hat{I} F$, " $h=1..p-1, g_j(b_{h+1}) - p_j(b_{h+1}) \geq g_j(b_h) + p_j(b_h)$ "

In order to define "distinguishable" categories, it is reasonable to impose that no alternative can be indifferent to more than one profile, i.e., " $a\hat{I} A$, " $h=1..p-1, alb_h \Rightarrow$ [not alb_{h+1} and not alb_{h-1}]" (a situation in which alb_h and alb_{h+1} would implicitly mean that the category delimited by the profiles b_h and b_{h+1} is "insufficiently wide"). Condition 2 is a sufficient condition for the preceding property to hold. In other words, it is possible to run ELECTRE TRI with profiles that do not fulfill condition 2 but in such cases some alternatives can be indifferent to two consecutive profiles.

3. ELECTRE-TRI Assistant : Assistance in the construction of a preference model Support for Parameter Elicitation

3.1 Inference of preference parameters

One of the main difficulties that an analyst must face when interacting with a DM in order to build a decision aid procedure is the elicitation of various parameters of the DM's preference model. In the ELECTRE TRI method, the analyst should assign values to profiles, weights and thresholds (see section 2). Even if these parameters can be interpreted, it can be difficult to fix directly their values and to have a clear global understanding of the implications of these values in terms of the output of the model.

[Mousseau & Slowinski 98] proposed a methodology that encompasses this problem by substituting assignment examples for direct elicitation of the model parameters. The values of the parameters are inferred through a certain form of regression on assignment examples. ELECTRE TRI Assistant implements this methodology in a way that requires from the DM much less cognitive effort: the elicitation of parameters is done indirectly (rather than directly) using holistic information given by the DM through assignment examples, i.e., alternatives assigned by the DM according to his/her preferences.

Assuming that a specific subset of parameters (possibly all of them) is to be optimised, mathematical program infers the values for these parameters that best restitutes the assignment examples (the general form of the mathematical program to be solved is given in appendix A). This is done in the course of an interactive process that gives to the DM the possibility to revise his/her assignment examples and/or change the set of parameters to be determined and/or to give additional information before the optimisation phase restarts.

The general scheme of this inference procedure using the paradigm of disaggregation is presented in Figure 10. Its aim is to find an ELECTRE TRI model as compatible as possible with the assignment examples given by the DM. The assignment examples concern a subset $A^* \subseteq A$ of alternatives for which the DM has clear preferences, i.e., alternatives that the DM can easily assign to a category, taking into account their evaluation on all criteria. The compatibility between the ELECTRE TRI model and the assignment examples is understood as an ability of the ELECTRE TRI method using this model to reassign the alternatives from A^* in the same way as the DM did.

3.2 Interactive learning process

In order to minimise the differences between the assignments made by ELECTRE TRI and the assignments made by the DM, an optimisation procedure is used. The resulting ELECTRE TRI model is denoted by M_{π} . The DM can tune up the model in the course of an interactive procedure (see Figure 10). He/she may either (1) revise the assignment examples or (2) change the set of parameters to be optimised or (3) fix values (or intervals of variation) for some model parameters. In the first case, the DM may:

- remove and/or add some alternatives from/to A^* ,
- change the assignment of some alternatives from A^* .

In the second case, he/she may remove and/or add some parameters from the set of those that are to be optimised.

In the last case, the DM can give additional information on the range of variation of some model parameters basing on his/her own intuition. For example, he/she may specify:

- ordinal information on the importance of criteria,
- noticeable differences on the scales of criteria,
- incomplete definition of some profiles defining the limits between categories.

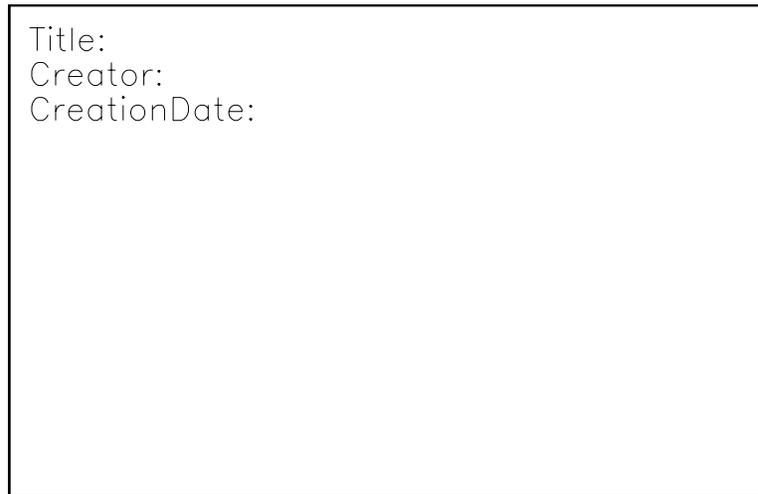


Figure 10: Inference scheme with ELECTRE TRI Assistant

When the model is not perfectly compatible with the assignment examples, the procedure is 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 is then asked to reconsider his/her judgement.

To get a representative model, the subset A^* must be defined such that the numbers of alternatives assigned to the categories are almost equal and sufficiently large to "contain enough information". The empirical behaviour of the inference procedure has been studied in [Naux 96] and [Mousseau *et al.* 97]. These experiments show that $2m$ (m being the number of criteria) is a sufficient number of assignment examples to infer the weights (the other parameters being fixed). Moreover, these studies proves the inference procedure to be suited for the interactive process described above.

The approach used in ELECTRE TRI Assistant is concordant with aggregation-disaggregation paradigm used for the construction of a preference model in UTA-like procedures (see [Jacquet-Lagrèze & Siskos 82], [Siskos & Yanacopoulos 85], [Yanacopoulos 85], [Jacquet-Lagrèze *et al.* 87], [Jacquet-Lagrèze 90], [Nadeau *et al.* 90], [Slowinski 91]). It has been also applied for the elicitation of weights used for the construction of an outranking relation in the DIVAPIME method (see [Mousseau 95] and [Mousseau 93]).

In order to infer the parameters of Electre Tri pessimistic assignment procedure (without veto) from assignment examples, optimization problem to be solved is the following:

$$\begin{aligned}
& \text{Max} \left(a + e \sum_{a_k \in A^*} (x_k + y_k) \right) \\
& \text{s.t.} \quad \sum_{j \in F} k_j c_j(a_k, b_{h_k-1}) / \sum_{j \in F} k_j - a - I \geq 0, \quad \forall a_k \in A^* \\
& \quad \quad I - \sum_{j \in F} k_j c_j(a_k, b_{h_k-1}) / \sum_{j \in F} k_j - a \geq 0, \quad \forall a_k \in A^* \\
& \quad \quad I \in [0.5, 1] \\
& \quad \quad g_j(b_{h+1}) \geq g_j(b_h) + p_j(b_h) + p_j(b_{h+1}), \quad \forall j \in F, \quad \forall h \in B \\
& \quad \quad p_j(b_h) \geq q_j(b_h), \quad \forall j \in F, \quad \forall h \in B \\
& \quad \quad k_j \geq 0, \quad q_j(b_h) \geq 0, \quad \forall j \in F, \quad \forall h \in B
\end{aligned}$$

where x_k and y_k represents slack variables whose meaning is such that all alternatives from the reference set A^* are "correctly" assigned for all $I' \in [I - \min_{a_k \in A^*} (y_k), I + \min_{a_k \in A^*} (x_k)]$. When $\varepsilon=0$, any non-negative value for the objective function guarantees existence of model parameters that permits "correct" assignment of all alternatives from A^* .

The above general problem is a non-linear programming problem. For n , m and p denoting the number of assignment examples, of criteria and of profiles, respectively, this problem contains $3mp+m+2$ variables and $4n+3mp+2$ constraints.

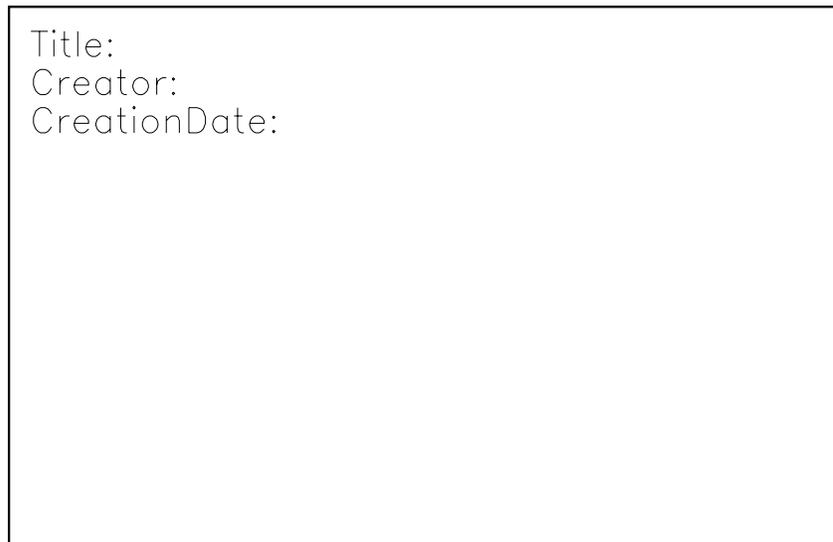
The optimization problem becomes linear when optimization is limited to the inference of weights.

When additional preference information on dependencies among the weights or on the range of their variation is given, additional constraints should be considered in the above formulation.

4. Software commands

4.1 Use of the software in a decision aid process

Real world decision processes are never sequential; the different phases in the definition of an assignment model interact (for example, the assignment of some alternative may reveal the necessity to consider an additional criterion). However, the general scheme of a decision aid process using the ELECTRE TRI method can be presented as shown in Figure 11.



Title:
Creator:
CreationDate:

Figure 11: General scheme of use of the ELECTRE TRI method

4.2 General organisation

The structure of the main menu available in the ELECTRE TRI 2.0a software is described in Figure 12. The content of the different options are the following:

- **File:** this option allows the user to create a new project, load an existing project and save the current project. Additional print and import options are provided.
- **Edit:** enables the user to enter the data required by ELECTRE TRI (criteria, alternatives, weights, profiles and thresholds) and/or to use the ELECTRE TRI assistant commands.
- **Results:** allows the user to visualise the results (including intermediary results such as degree of credibility of the outranking relation, comparison of alternatives to profiles,...); also gives a graphical representation of alternatives and profiles.
- **Windows:** gives the possibility to manage the appearance of the windows on the screen.
- **Help:** provides the user an online help.

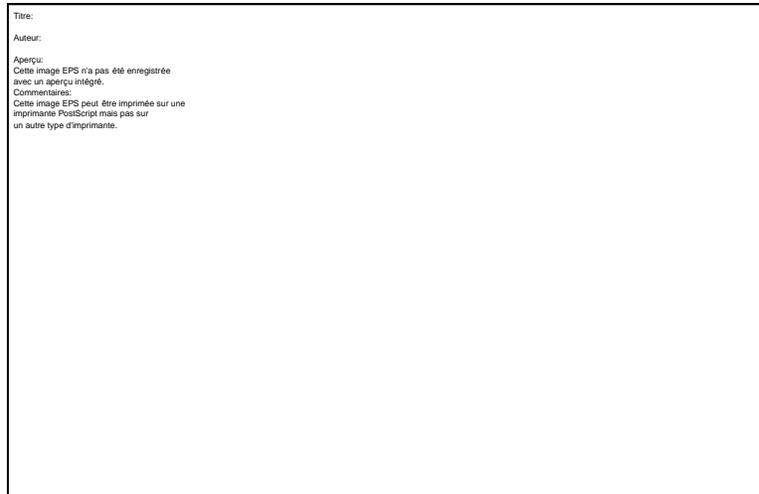


Figure 12: Structure of the software options

4.3 The File menu commands

4.3.1 New project

This command allows to create a new project. This new project will be associated to a new data set that you will have to create. This command leads you to the **Edit Project** Dialog box in which the data concerning the project can be entered.

The first button in the Tool bar  is a short-cut for this command.

4.3.2 Open project

This command may be used to load in memory a data set created during a previous session of ELECTRE and that has been saved on disk.

The second button in the Tool bar  is a short-cut for this command.

You have to type the name of the project or to select it in the file list. You may choose the drive on which your file is saved and the directory in the **Directories** list box window. The **Files of type** combo box gives a list of all files that have the mask proposed in the **File Name** list box window. By default, ELECTRE gives a list of the files having the extension .BDF and .ELP in the current directory.

It is also possible to import data from the previous version of ELECTRE TRI using the **Files of type** combo box, and selecting *<Electre Tri 1.0 files>*.

4.3.3 Close project

This command may be used to remove from memory the current data set. If you want to keep your dataset, you must use the **Save project** command before using this command.

However, if the current project was changed, the **Save project as** dialog box will be displayed automatically.

4.3.4 **S**ave project

This command allows to save the project currently in memory with its current name. It can only be used when a project has previously been created or loaded.

The third button in the Tool bar  is a short-cut for this command.

If the project has just been created, ELECTRE TRI displays the dialog box **Save Project As** so that you may give a name to your project.

4.3.5 **S**ave project **A**s

This command allows to save the current project under a name different from its current name or to save a project for the first time.

To save a project with its current name, you should use the **Save project** command. Choose the drive and the directory in the **Directories** list box window and type the name of the file in the **File Name** list box window. If you do not give any extension to the file name, ELECTRE TRI will add the extension .BDF. If you type an existing file name in the chosen directory, ELECTRE TRI will ask confirmation before removing the existing file.

4.3.6 **I**mport alternatives from a text file

This command may be used to import the data (only the alternatives) from the an ASCII file in a specific format (see also § 5.8 for more information). The text file contain on each of its lines the information concerning a single alternative with the following format:

- the name of the alternative (string up to 8 characters without any spaces) in the first column,
- column separator (<TAB> character),
- the description of the alternative (string up to 255 characters without any spaces) in the second column,
- column separator (<TAB> character),
- the performances of the alternative (numeric values between -999999 and +999999 with a dot (.) as decimal separator) in the next columns. Each performance should be separated by the <TAB> character.

4.3.7 Project Report

This command may be used to generate a report (.RPT ascii file) on the current project. This file contains the following information :

- list of criteria,
- list of profiles and corresponding thresholds,
- list of all alternatives.

4.3.8 Print project

This command allows to print all or part of the data and/or the results. You have to select the elements you wish to print.

The fourth button in the Tool bar  is a short-cut for this command.

Use the **Print Setup** command to choose the printer and to define printing parameters. You may also print in a file (see **Print setup**).

4.3.9 Print setup

This command may be used to choose the printer and to define printing parameters such as the orientation (Portrait or Landscape). These parameters depend on the selected printer. The printers that are displayed are those installed with Windows. To add new printers, you have to use the **Control Panel** of Windows.

4.3.10 Exit

This command closes ELECTRE TRI. You also may double-click on the System box of the window ELECTRE TRI or type ALT-F4.

If the current project has been modified since it has last been saved on disk, ELECTRE TRI will ask if you would like to save before exiting.

4.4 The Edit menu commands

4.4.1 The Edit Data set command

This command may be used to visualise and/or modify all data related to the project, i.e., information concerning the owner and description of the project, the criteria, the profiles defining the categories, the alternatives.

The fifth button in the Tool bar (displaying a folder) is a short-cut for this command.

The  button enables the user to insert a criterion, a profile or an alternative (according to the element selected on the left part of the dialog window).

The  button enables the user to delete the criterion, the profile or the alternative selected on the left part of the dialog window.

The  button is used to exit the **Edit Project** dialog window.

The **Edit Project** dialog window is composed of two parts.

- The left part describes the list of data to be entered/modified. The lists of criteria, profiles and alternatives can be open and closed by clicking on the (+) and (-) buttons. A specific criterion, profile or alternative can be disabled/enabled by a double click.
- The right part of the dialog window enables the user to enter/modify the data that is selected in the left part of the dialog window. The right part of the dialog window is composed of folders in which different information can be entered/modified.

4.4.1.1 Editing General Information

When selecting **Project** on the left side of the **Edit Project** dialog window, you can edit:

- a text description of the project and the name of the owner in the information folder,
- the cutting level λ in the method folder.

4.4.1.2 Editing Criteria

When selecting **Criteria** on the left side of the **Edit Project** dialog window, two folders are available:

- the information folder specifies the total number of criteria, the number of defined criteria (those completely defined) and the number of enabled criteria.

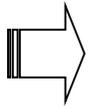
 button give the possibility to make all criteria active.

- the weight folder enables to edit the weights of all criteria.

When selecting a specific criteria in the list (on the left part of the **Edit Project** dialog window), two folders are available :

- the definition folder enables to edit the name, code, weight and direction of preference of the selected criterion,
- the performances folder enables to edit the performances (evaluations) of the alternatives on the selected criterion.

4.4.1.3 Editing Profiles



Important remark: the profiles must be entered from the best (profile defining the limit between the best and the second best category) to the worst (profile defining the limit between the worst and the second worst category)

When selecting **Profiles** on the left side of the **Edit Project** dialog window, two folders are available:

- the information folder specifies the total number of profiles, the number of defined profiles (those completely defined) and the number of enabled profiles. The  button give the possibility to make all profiles active.
- the categories folder enables to edit the names of the categories defined by the limit profiles.

When selecting a specific profile in the list (on the left part of the **Edit Project** dialog window), three folders are available :

- the definition folder enables to edit the name and code of the selected profile,
- the performances folder enables to edit the performances (evaluations) of the selected profile on all criteria,
- the thresholds folder enables to edit the indifference, preference and veto threshold attached to the selected profile (for each criterion).

4.4.1.4 Editing Alternatives

When selecting **Alternatives** on the left side of the **Edit Project** dialog window, one folder is available:

- the information folder specifies the total number of alternatives, the number of defined alternatives (those completely defined) and the number of enabled alternatives. The  button give the possibility to make all alternatives active.

When selecting a specific alternative in the list (on the left part of the **Edit Project** dialog window), two folders are available :

- the definition folder enables to edit the name and code of the selected alternative,
- the performances folder enables to edit the performances (evaluations) of the selected alternative on all criteria,

4.4.2 ELECTRE TRI Assistant

This command provides support to the user in the definition of the values of parameters of the assignment model. The software provides the possibility to be supported when defining the weights of criteria and the cutting level. A step by step example is provided in section 5.9. The use of ELECTRE TRI Assistant functionalities proceeds according to the following scheme:

1. Input a list of assignment examples composed of alternatives for which the DM gives a holistic assignment (such alternative can be an existing alternative of a fictitious one designed for this purpose). Imprecise assignments are accepted, i.e., the DM can express an hesitation in the assignment of an alternative *a* by specifying a subset of consecutive categories to which *a* could be assigned.
2. Give eventually preferential information on the weights,
3. Run the inference procedure to find values for the parameters,
4. Check for the acceptability of the obtained weights and either:
 - accept the proposed weights so as to use it as an assignment rule,
 - or reject it and revise the information stated in the step 1. and 2. in order to perform step 3 again.

The ELECTRE TRI Assistant command leads to a menu with different options proposed in a menu screen. The structure of these options is described in Figure 13.

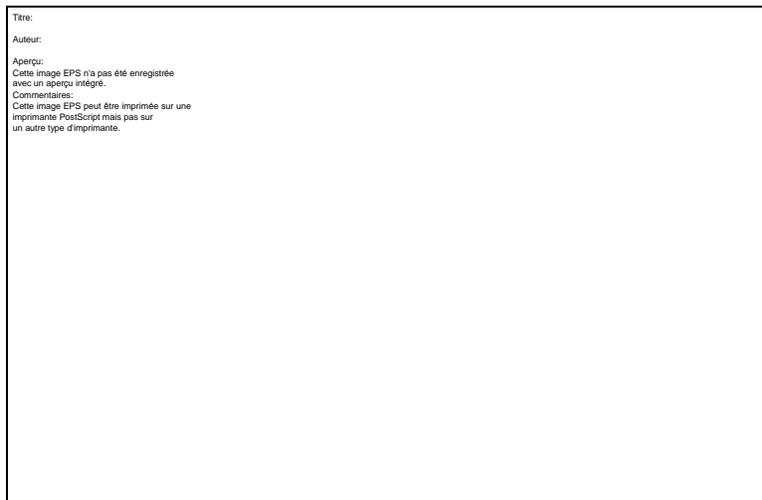


Figure 13: ELECTRE TRI Assistant commands

4.4.2.1 List of assignment examples command

This command enables to edit the list of assignment examples (i.e., alternatives for which the user gives a holistic intuitive assignment), and is available through the

 button.

This command leads to the List of assignment examples dialog window in it is possible to:

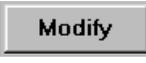
- add an assignment example corresponding to an existing alternative ( button), i.e., select an existing alternative from the list and

specify the minimum and maximum category to which this alternative should be assigned,

- add an assignment example grounded on a fictitious alternative ( button), i.e., edit the performances of the fictitious alternative and specify the minimum and maximum category to which this alternative should be assigned,
- delete an existing assignment example ( button),
- modify an existing assignment example (Edit button),
- validate the modification and return to the ELECTRE TRI Assistant menu ( button),
- cancel all modifications ( Button).

4.4.2.2 Preference information on weights command

This command can be used to edit additional preferential information on the weights on the Preference information on weights. Such additional information can be :

1. a ranking of criteria according to their relative importance (this is done using the  button through a specific interface window),
2. some comparisons of coalition of criteria according to their relative importance (such comparisons can be added modified or deleted using the ,  and  buttons),
3. lower and upper bound for the weights (use the  button).
- 4.

The user can then validate the modification and return to the ELECTRE TRI Assistant menu menu ( button), or cancel all modifications ( Button).

4.4.2.3 Preference information on cutting level command

This command can be used to edit additional preferential information on the cutting level λ . Such information take the form of lower and upper bound for the cutting level λ . This command is available through the  button on the ELECTRE TRI Assistant menu.

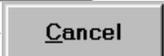
4.4.2.4 Infer weights command

This command launches the computations necessary to infer the weights that best match to the assignment examples. Note that this command is available only when assignment examples (at least one) have been input; if not the infer weight button remains grey and is not effective.

It leads to the Preview assistant data window that synthetizes the Electre Tri Assistant input data. From this window, the user may go back to the Assistant menu ( button), or continue the computations ( button).

At the end of the computations, the results are presented in the Optimization results window. For each assignment example, the following information is presented:

- the category to which the alternative should be assigned,
- the category to which the alternative is assigned using the computed weights.

The user can then go back to the Electre Tri Assistant menu in order to modify the input data ( button) or see the computed weights ( Button). In the latter case, the user can then accept the computed weights and exit from Electre Tri Assistant ( button) or go back to the Electre Tri Assistant menu in order to modify the input data ( button).

4.4.2.5 Load Data command

This command may be used to load in memory an assistant data set created during a previous session of ELECTRE TRI Assistant and that has been saved on disk (.ETA file).

You have to type the name of the .ETA file or to select it in the file list. You may choose the device on which your file is saved and the directory in the window Directories. The window Files gives a list of all files that have the mask proposed in the window File Name. By default, ELECTRE TRI gives a list of the files having the extension .ETA in the current directory. This command is available using the  button in the ELECTRE TRI Assistant Menu.

Important remark: the chosen .ETA file should correspond to the current project in memory.

4.4.2.6 Save data command

This command enables to save the current assistant data (list of assignment examples, preferential information on weights and/or cutting level) into a specific .ETA file.

This command is available using the  button in the ELECTRE TRI Assistant Menu.

4.4.2.7 Exit command

This command enables to exit from the ELECTRE TRI Assistant sub-menu. Remember to save your Assistant data (in a .ETA file using the Save data command).

4.5 The Result menu commands

4.5.1 Assignment by Category

This command displays the results of the two assignment procedures (optimistic and pessimistic) for all alternatives of the current project. All alternatives assigned to the selected category are grouped together. The button  in the tool bar is a shortcut to this command.

4.5.2 Assignment by Alternative

This command displays the results of the two assignment procedures (optimistic and pessimistic) for all alternatives of the current project. The assignment of all alternatives are presented consecutively in a table. The button  in the tool bar is a shortcut to this command.

4.5.3 Comparison to profiles

This command displays the intermediary results concerning the comparisons of all alternatives to each limit profile. The symbols used are **f** (or **p**) for a preference, **I** for indifference and **R** for incomparability.

4.5.4 Performances of alternatives

This command displays the performance table (evaluation matrix), i.e., the performance of each alternative on each criterion.

4.5.5 Degrees of credibility

This command displays the intermediary results concerning the degree of credibility of the outranking relation between each profile and each alternative.

4.5.6 Visualisation of alternative

This command provides a visual representation of an alternative and the profiles. The representation enable to visualize one alternative at a time (in red) together with the profiles specifying the limits of the categories (in blue).

4.5.7 Statistics of assignment

This command gives a statistical synthesis of the assignments, i.e., the proportion of alternatives assigned to each category for both assignment rules (optimistic and pessimistic).

4.6 The Window menu commands

4.6.1 Cascade

This command may be used to organise the different windows open on the desktop in such a way that their titles are always visible despite overlapping.

4.6.2 Tile

This command may be used to organise the different windows open on the desktop in such a way that they are all entirely visible.

4.6.3 Arrange Icons

This command may be used to reorganise the different icons on the desktop, they will be gathered at the bottom of the screen

4.6.4 Close all

This command closes all windows and icons open on the desktop.

4.7 Help menu commands

4.7.1 Contents

This command displays a short help text on every available command in the software.

4.7.2 Search Topic

This command helps to quickly find the main procedures such as create, execute, read a project...

4.7.3 How to Use Help

This command displays a help information about using the Windows Help.

4.7.4 About

This command displays the version number and the copyright note.

5.A step-by-step example

5.1 Creating a new project

When beginning your first session with ELECTRE TRI 2.0, it is necessary to create a new project. This can be done by using the **New Project** option of the **File** command menu. A shortcut to this command is provided by the first button on the left .

This command lead to the *Edit Project* window with a blank project (see Screen 5.1).

5.2 Editing the data set

the *Edit Project* window allows the user to input/modify all data related to the project, i.e., information concerning the owner and description of the project, the criteria, the profiles defining the categories, the alternatives.

The **Edit Project** dialog window is composed of two parts.

- The left part describes the list of data to be entered/modified. The lists of criteria, profiles and alternatives can be open and closed by clicking on the  and  symbols. A specific criterion, profile or alternative can be disabled/enabled by a double click.
- The right part enables the user to enter/modify the data that is selected in the left part of the dialog window. The right part of the screen is composed of folders in which different information can be entered/modified.

The  button enables the user to insert a criterion, a profile or an alternative (according to the element selected on the left part of the window).

The  button enables the user to delete the criterion, the profile or the alternative selected on the left part of the window.

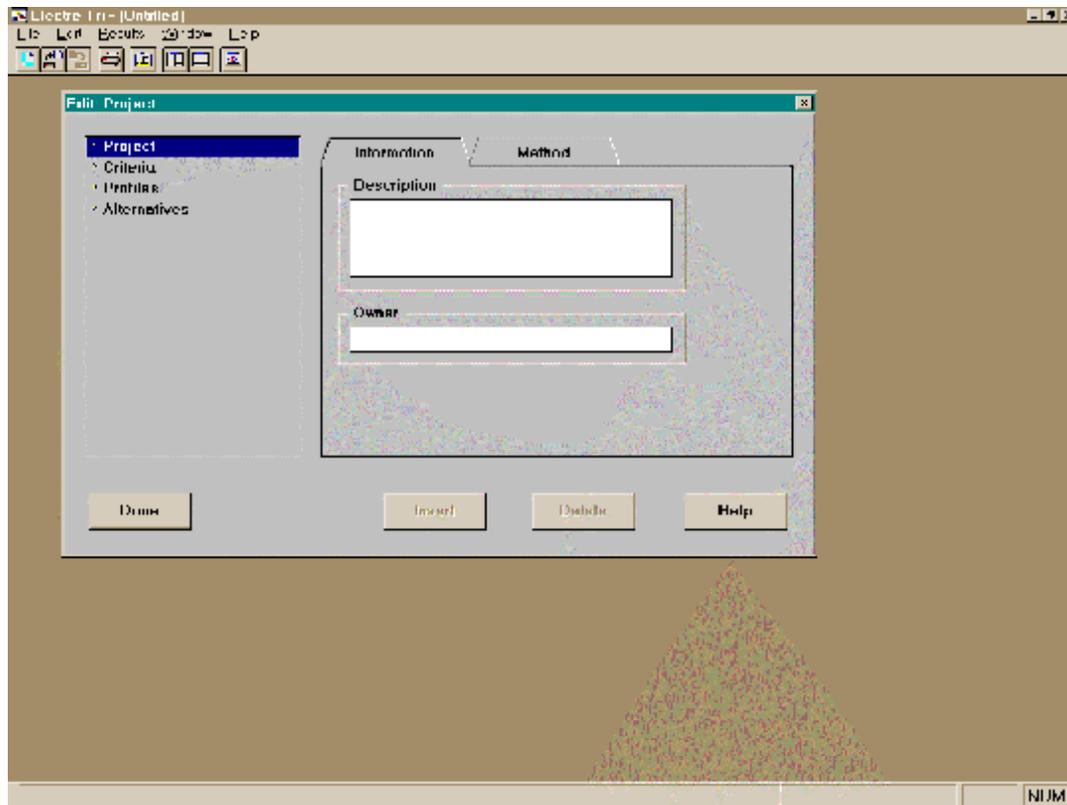
The  button is used to exit the **Edit Project** dialog window.

5.2.1 Editing General Information

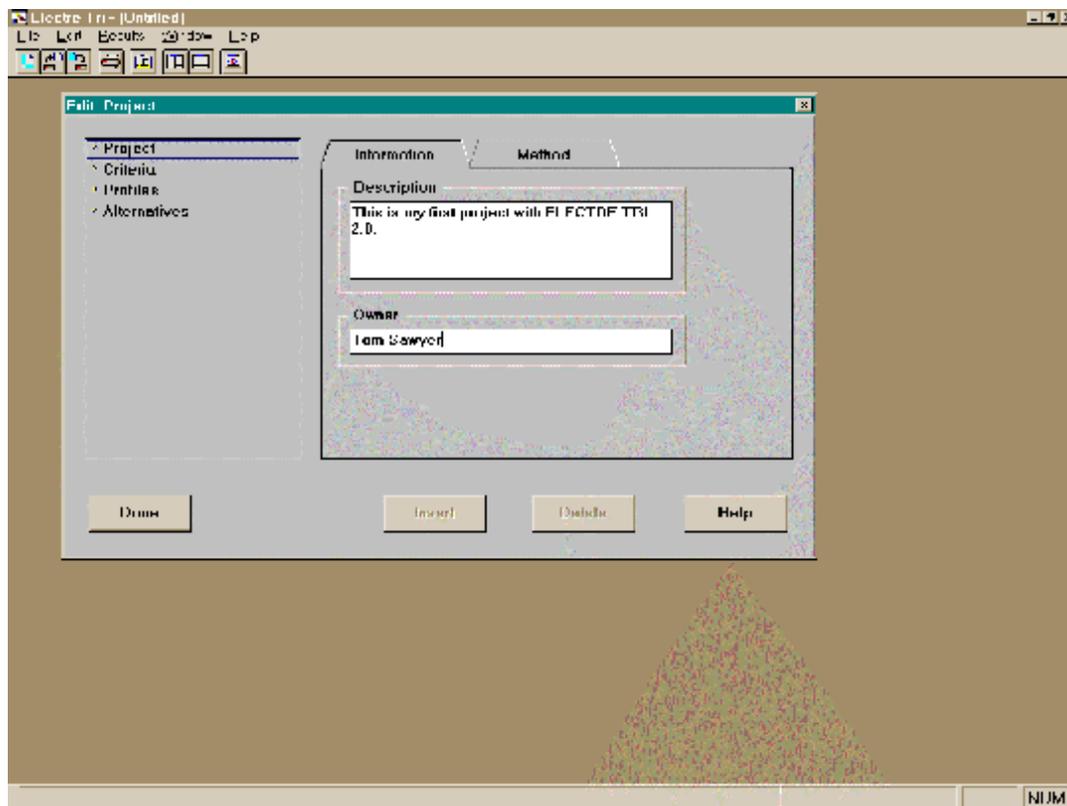
When selecting **Project** on the left side of the **Edit Project** dialog window, you can edit (see Screen 5.2):

- a text description of the project and the name of the owner in the information folder,

the cutting level λ in the method folder (see Screen 5.3).



Screen 5.1: Edit Project blank window



Screen 5.2: Edit General Information window

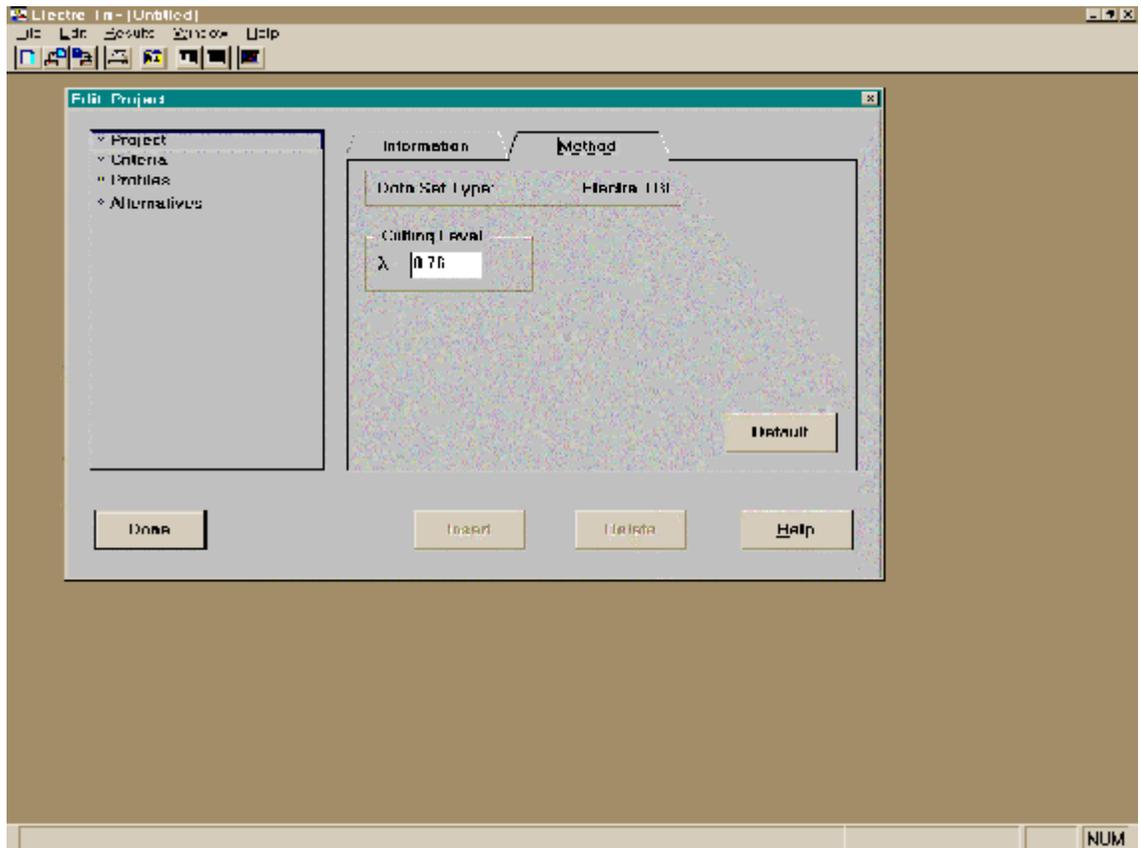
5.2.2 Inserting and editing criteria

In order to insert criteria in the project, it is necessary to click on <Criteria> on the left side of the window, and to click on the  button as many times as necessary (5 times for 5 criteria). Criteria appear in the list, on the left side of the screen (see Screen 5.4).

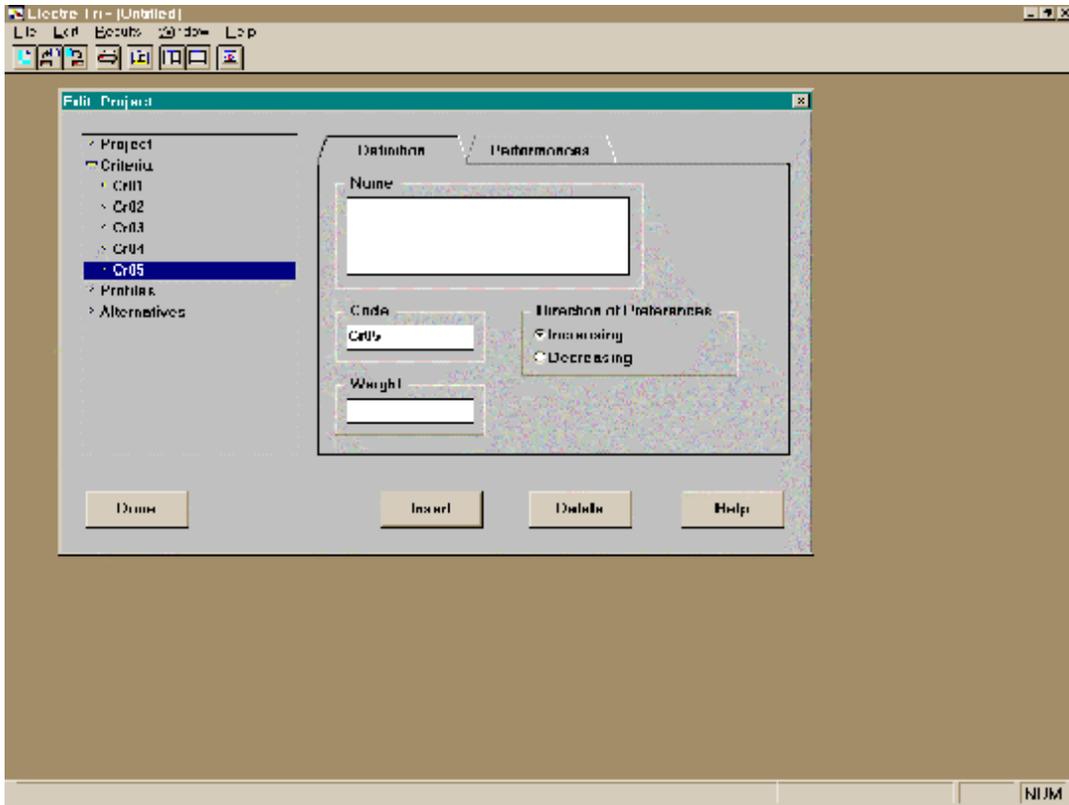
So as to input information concerning the inserted criteria, proceed as follows:

1. Select **Criteria** on the left side of the **Edit Project** dialog window, two folders are available (see Screen 5.5):

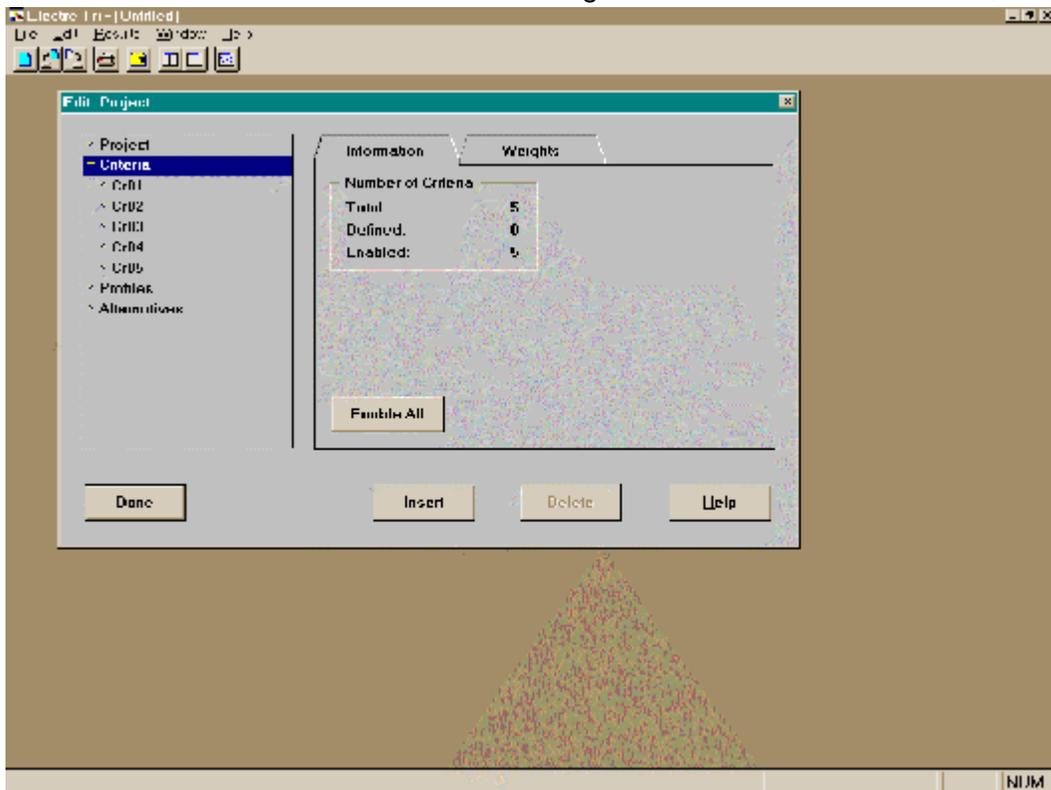
- the **information** folder specifies the total number of criteria, the number of defined criteria (those completely defined) and the number of enabled criteria. The  button give the possibility to make all criteria active (see Screen 5.5).
- the **weight** folder enables to input/modify the weights of all criteria (see Screen 5.6).



Screen 5.3: Edit cutting level folder



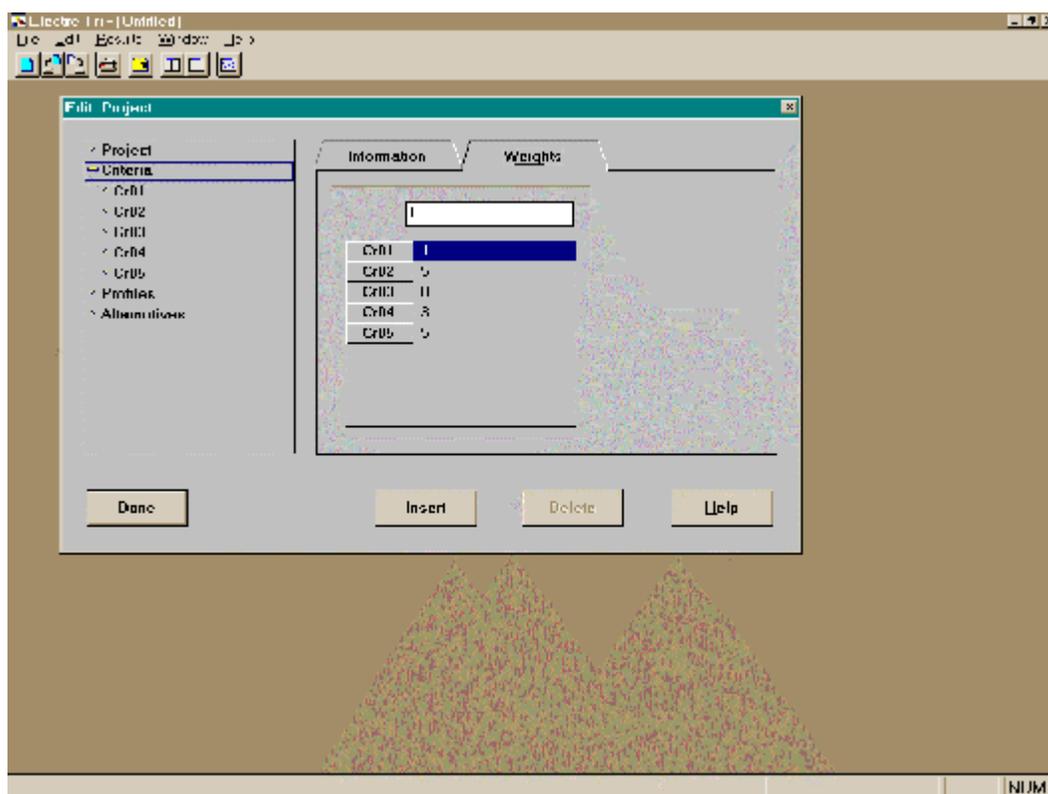
Screen 5.4: Inserting criteria



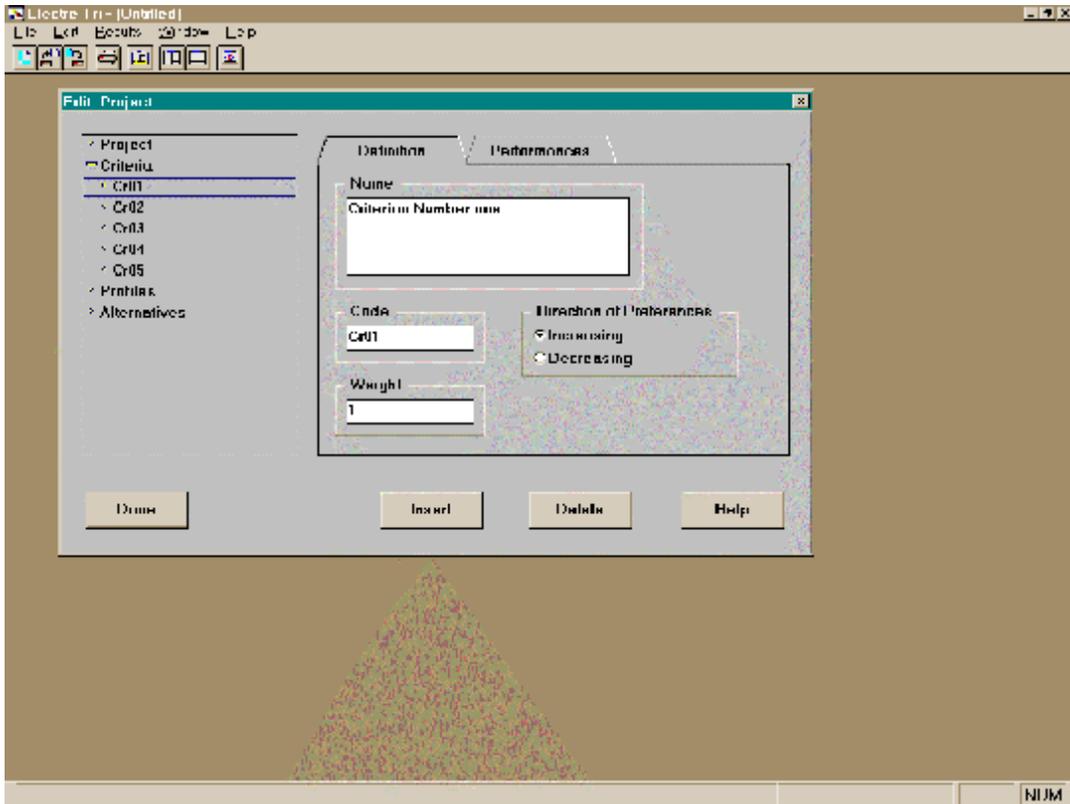
Screen 5.5 : Folder displaying general information about criteria

2. Select a specific criterion in the list (on the left part of the **Edit Project** dialog window), two folders are available :

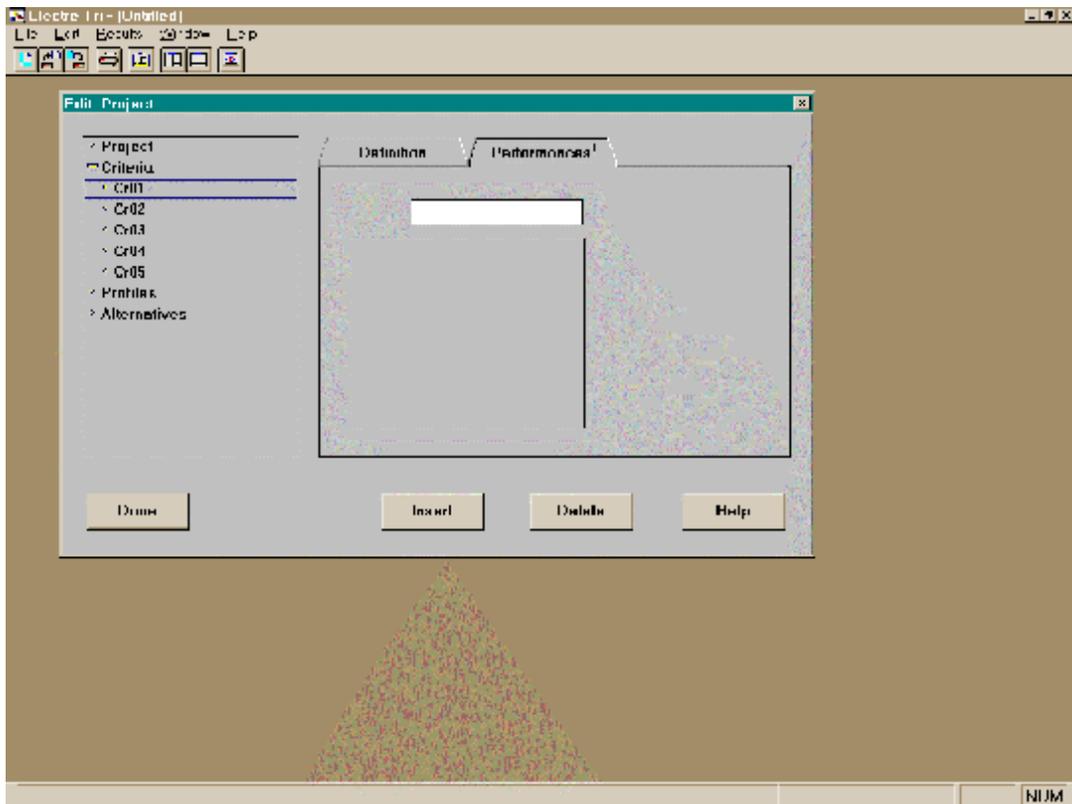
- the **definition** folder enables to edit the name, code, weight and direction of preference of the selected criterion (see Screen 5.7),
- the **performances** folder enables to edit the performances (evaluations) of the alternatives created (no alternative exist for the moment) on the selected criterion (see Screen 5.8).



Screen 5.6: Weights edit folder



Screen 5.7: Edit criterion definition Folder



Screen 5.8 : Edit performances of alternatives on a specific criterion

5.2.3 Inserting and editing profiles

In order to insert profiles in the project, it is necessary to click on <Profiles> on the left side of the window, and to click on the  button as many times as necessary (twice for 2 profiles, i.e., 3 categories). Profiles appear in the list, on the left side of the screen (see Screen 5.9).

**The first profile in the list should be the frontier between the best and second best category (default name Pr01),
the second profile should be the frontier between the the second best category and the one just below (default name Pr02),
....
the last profile corresponds to the frontier between the the worst category and the one just above.**

So as to input information concerning the inserted profiles, proceed as follows:

1. Select **Profiles** on the left side of the **Edit Project** dialog window, two folders are available (see Screen 5.10):

- the information folder specifies the total number of profiles, the number of defined profiles (those completely defined) and the number of enabled profiles. The  button give the possibility to make all profiles active (see Screen 5.10).
- the **categories** folder enables to input/modify the names of the categories of all criteria (see Screen 5.12).

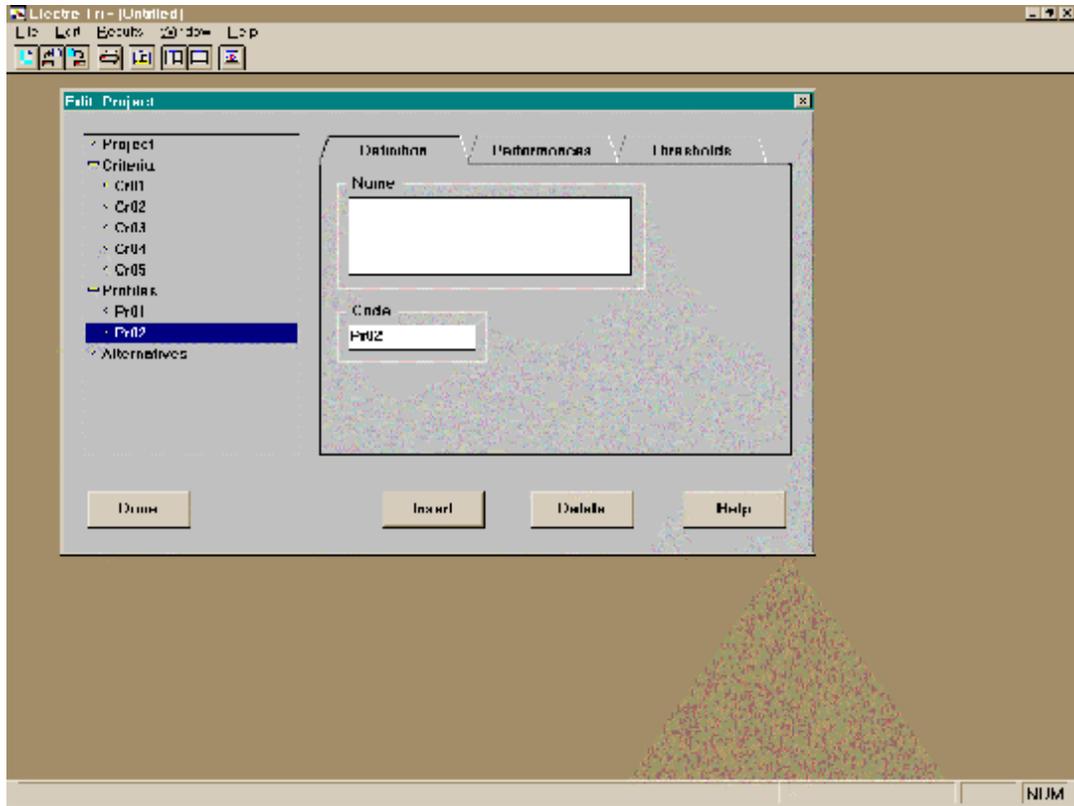
In the list, the categories must be ordered from the best to the worst

2. Select a specific profile in the list (on the left part of the **Edit Project** dialog window), three folders are available :

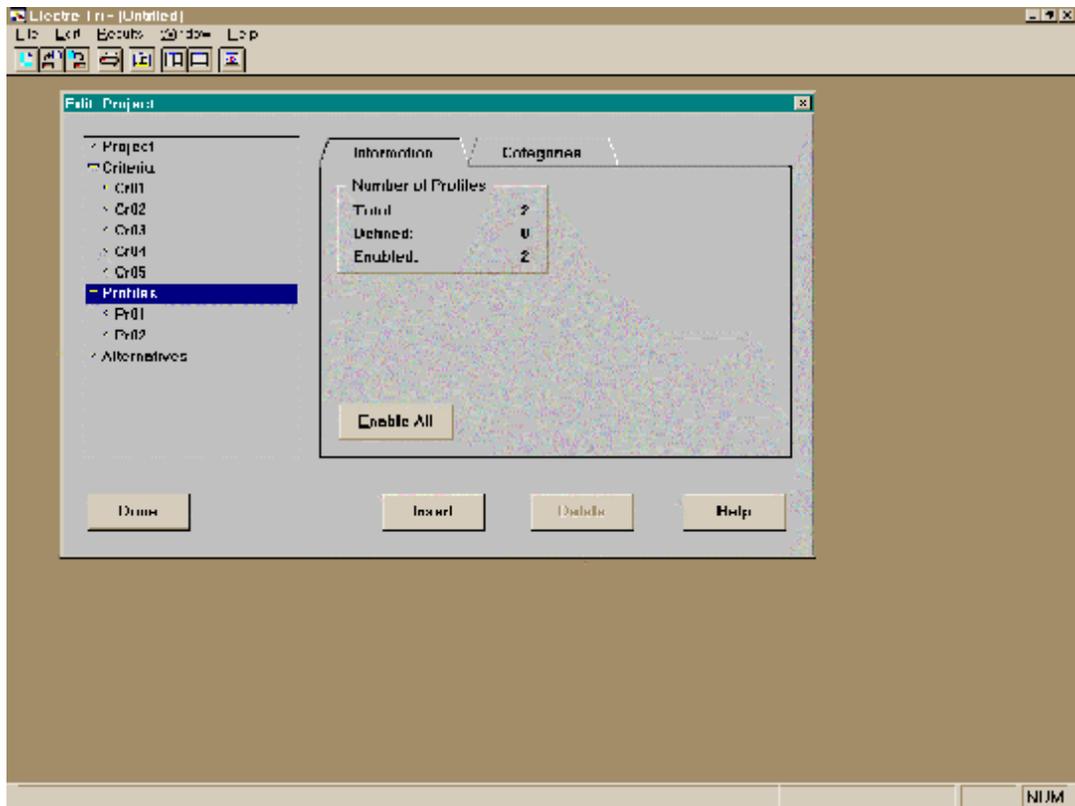
- the **definition** folder enables to edit the name and code of the selected profile (see Screen 5.13),
- the **performances** folder enables to edit the performances (evaluations) of the profile on all criteria. In order to do so, double click with the left button of the mouse on the cell corresponding to the value to be input (see Screen 5.14).

The performances of profiles on criteria should be consistent with the order of profiles in the list, i.e., ordered from the best to the worst.

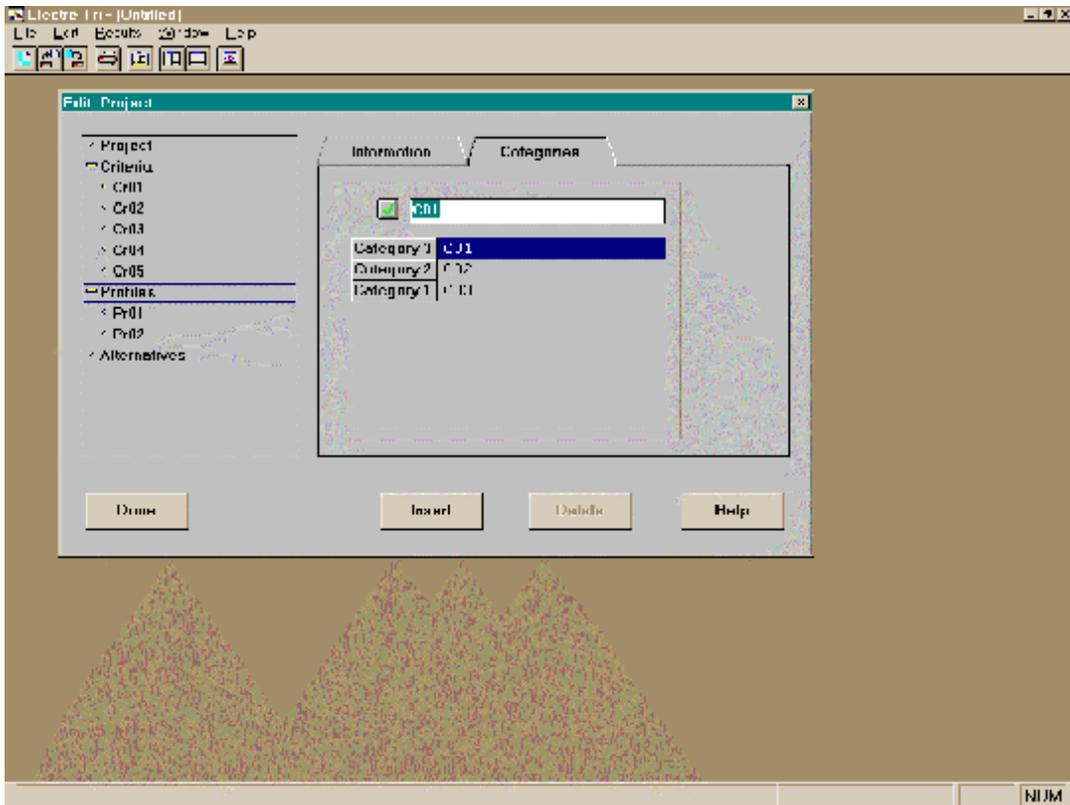
1. select the criterion in the list,
2. input the value for the indifference and preference thresholds in the corresponding boxes,
3. if you want to specify a veto threshold, click on the **Disable veto** box and input the value for the veto threshold.
4. Repeat step 1. to 3. for each criterion.



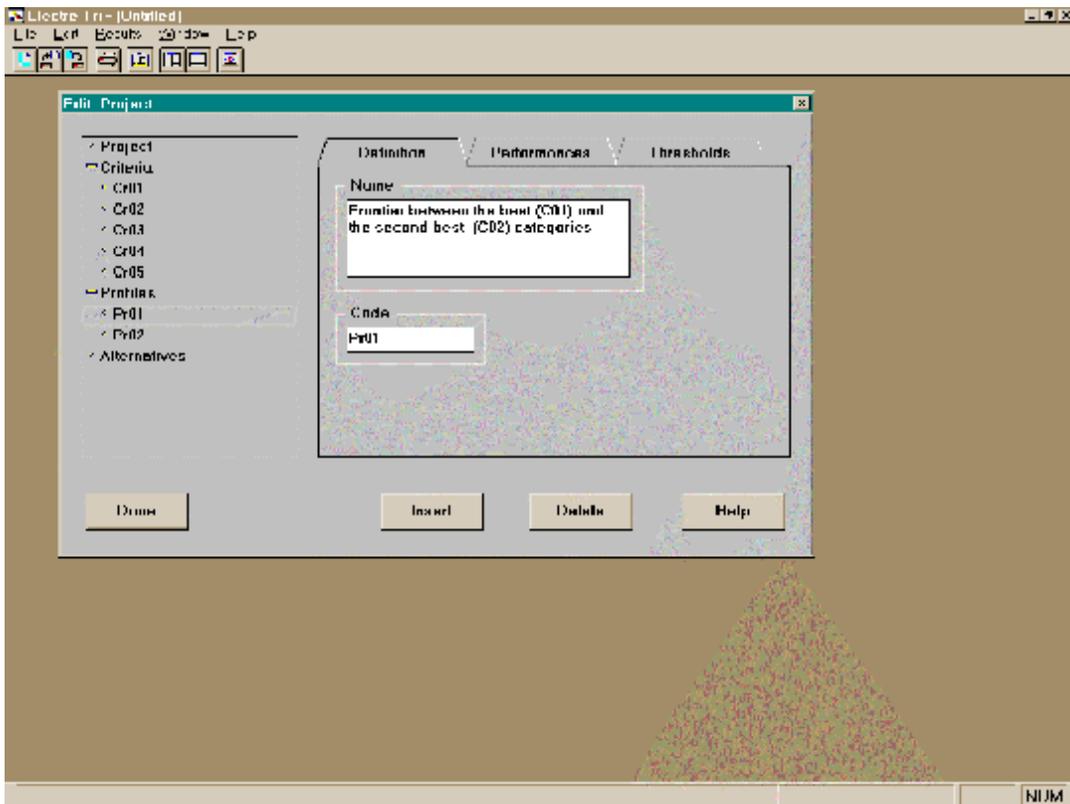
Screen 5.9 : Inserting profiles



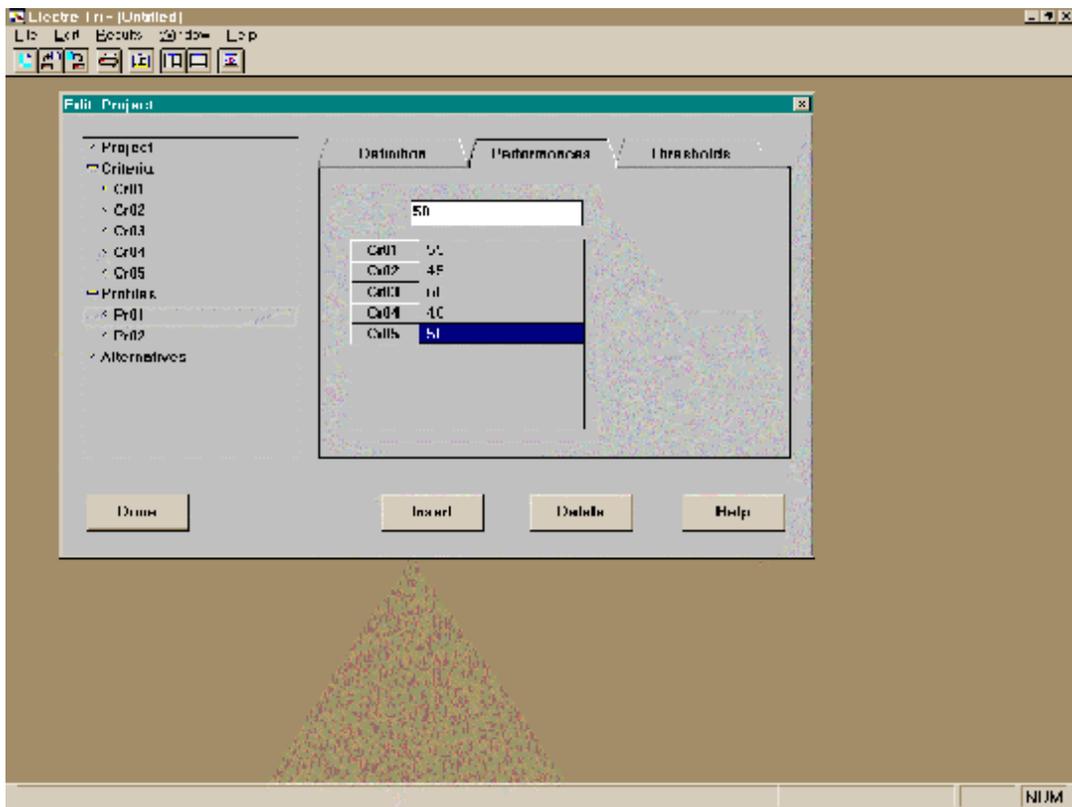
Screen 5.10 : General information concerning profiles Folder



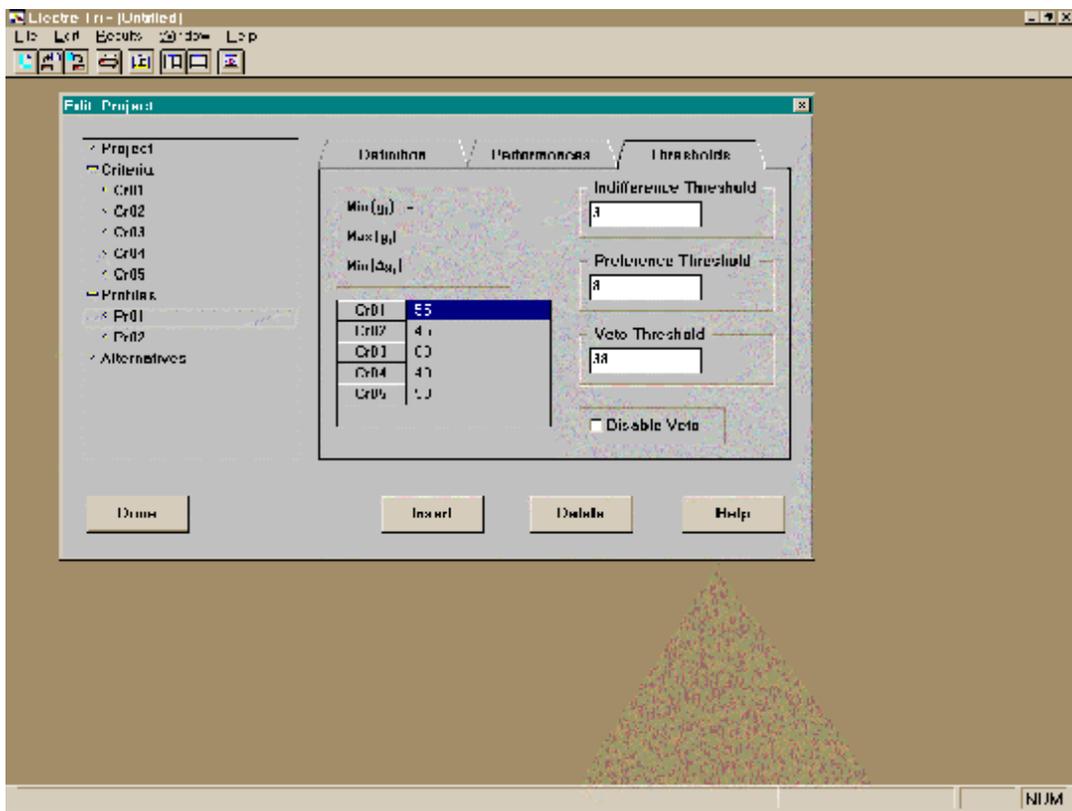
Screen 5.11: Specifying the names of the categories



Screen 5.12 : Specifying the name and code of a profile



Screen 5.13: Defining the performances of a profile



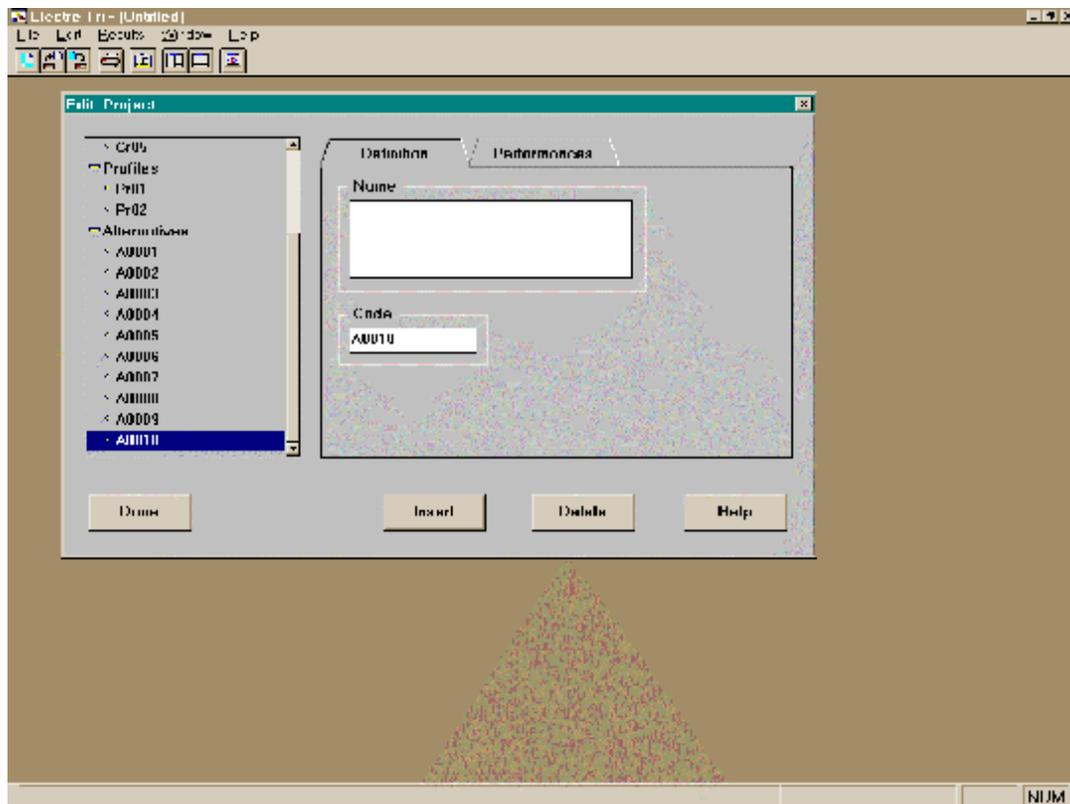
Screen 5.14: Defining the thresholds attached to a profile

5.2.4 Inserting and editing alternatives

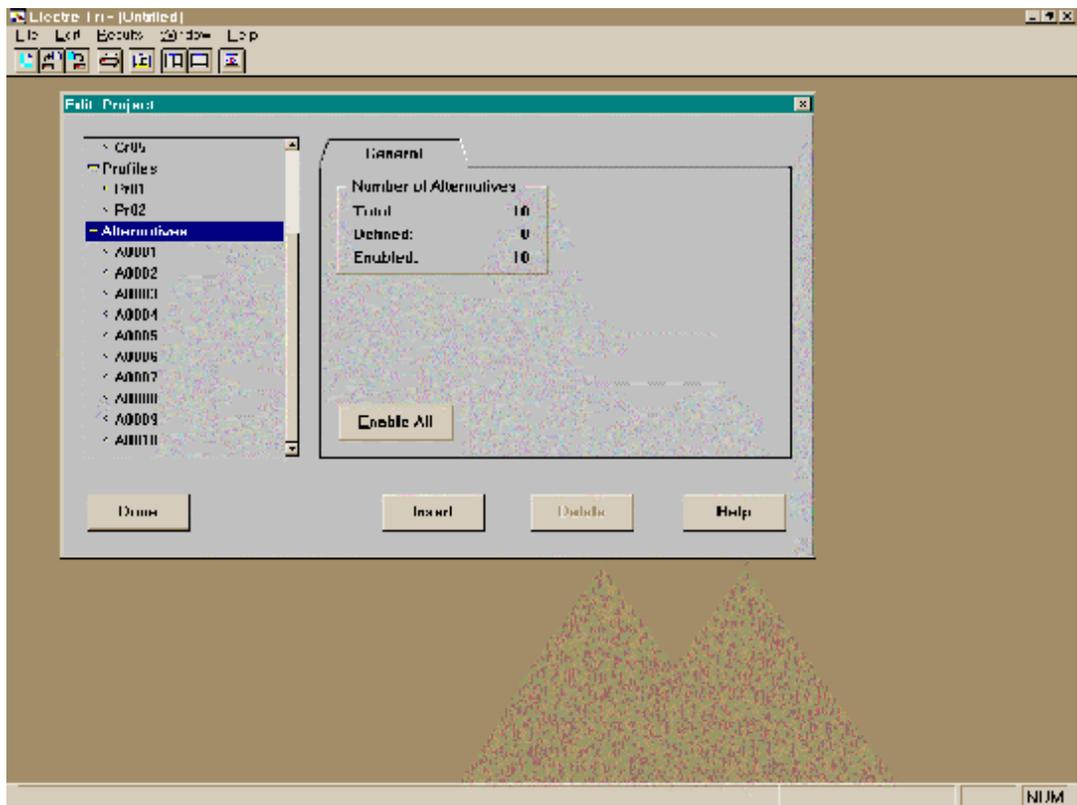
In order to insert alternatives in the project, it is necessary to click on <alternatives> on the left side of the window, and to click on the  button as many times as necessary (10 times for 10 alternatives). Alternatives appear in the list, on the left side of the window (see Screen 5.15).

So as to input information concerning the inserted alternatives, proceed as follows:

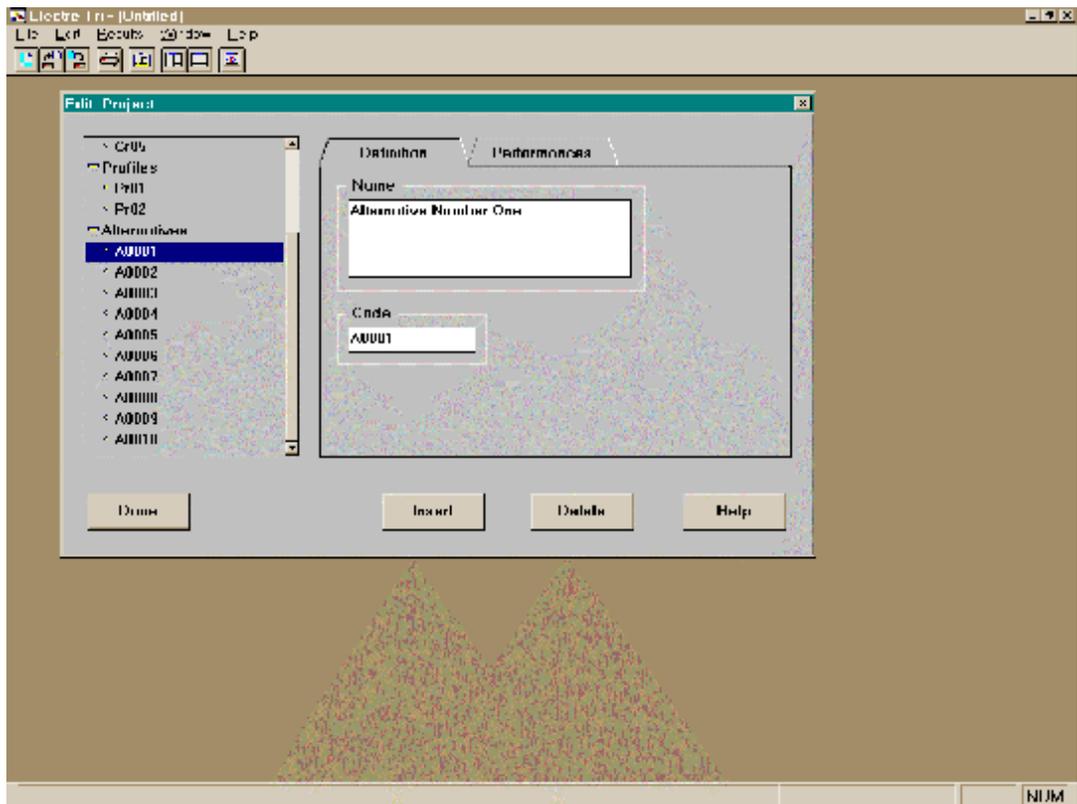
1. Select **Alternatives** on the left side of the **Edit Project** dialog window, one folder is:
 - the **General** folder specifies the total number of alternatives, the number of defined alternatives (those completely defined) and the number of enabled alternatives available (see Screen 5.16). The  button give the possibility to make all criteria alternatives.
2. Select a specific alternative in the list (on the left part of the **Edit Project** dialog window), two folders are available :
 - the **definition** folder enables to edit the name and of the selected alternative (see Screen 5.17),
 - the **performances** folder enables to edit the performances (evaluations) of the on all criteria (see Screen 5.18).



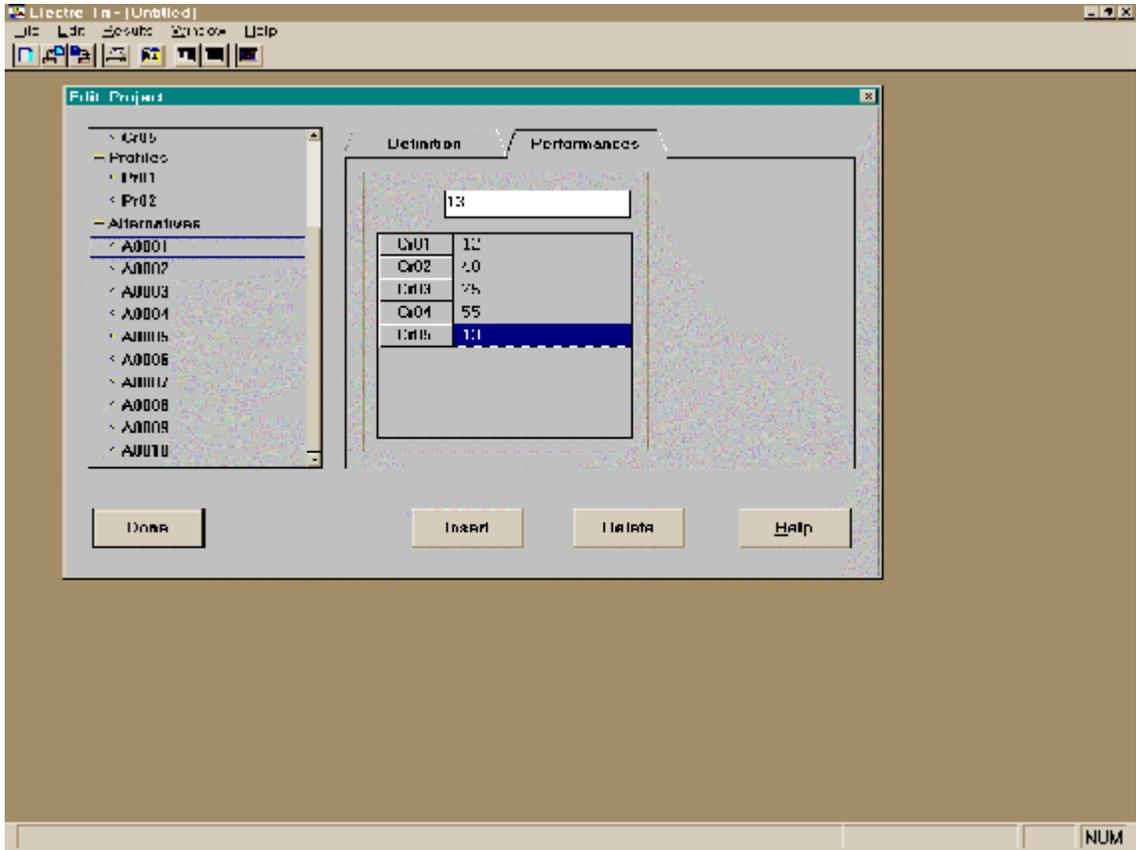
Screen 5.15: Inserting alternatives



Screen 5.16: General information concerning the alternatives



Screen 5.17: Editing the name and code of an alternative



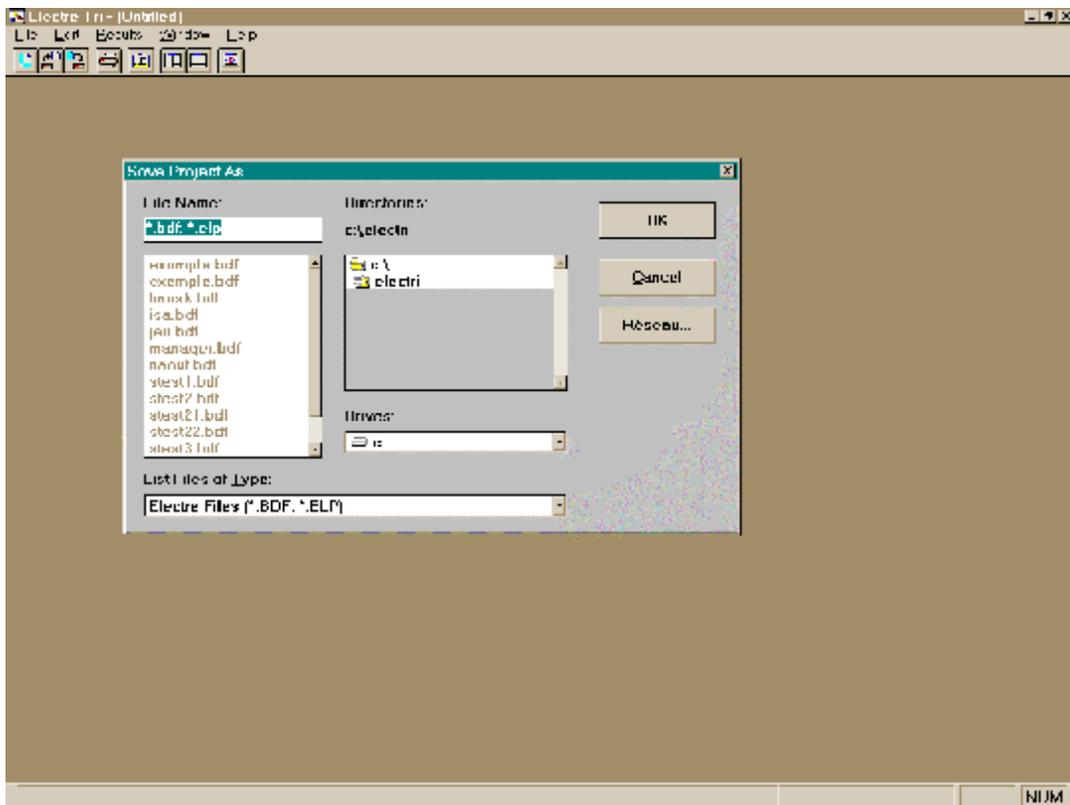
Screen 5.18: Specifying the performances of an alternative

5.3 Saving the data set

In order to save the project currently in memory, choose the **Save Project** option in the **File** command menu. As the project has just been created, ELECTRE TRI displays the dialog box **Save Project As** so that you specify your project (see Screen 5.19). Choose the drive and the directory in the window **Directories** and type the name of the file in the window **File Name** (example1.bdf). If you do not give any extension to the file name, ELECTRE TRI will add the extension .BDF. If you type an existing file name in the chosen directory, ELECTRE TRI will ask confirmation before removing the existing file.

The third button in the Tool bar  is a short-cut for this command.

If you want to save the current project under a name different from its current name or to save a project for the first time, choose the **Save Project As** option in the **File** command menu.



Screen 5.19: Saving the current project

5.4 Obtaining results

5.4.1 Assignment by category

In order to obtain the results of the two assignment procedures (optimistic and pessimistic) presented as the list of alternatives assigned to each category, select the **Assignment by Category** option in the **Result** command menu.

To see the alternatives assigned to a specific category, it is necessary to click on the corresponding category in the table (see Screen 5.20).

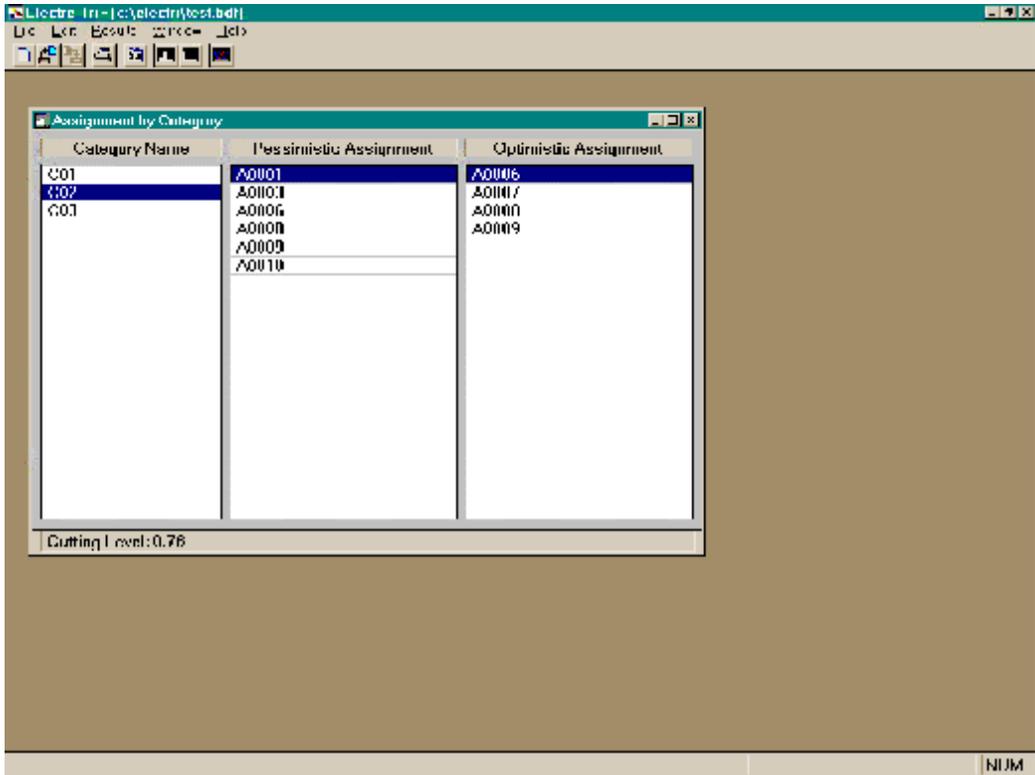
5.4.2 Assignment by alternative

In order to obtain the list of alternatives and their assignment by the two procedures (optimistic and pessimistic), select the **Assignment by Alternative** option in the **Result** command menu. The assignment of all alternatives are presented consecutively in a table (see Screen 5.21).

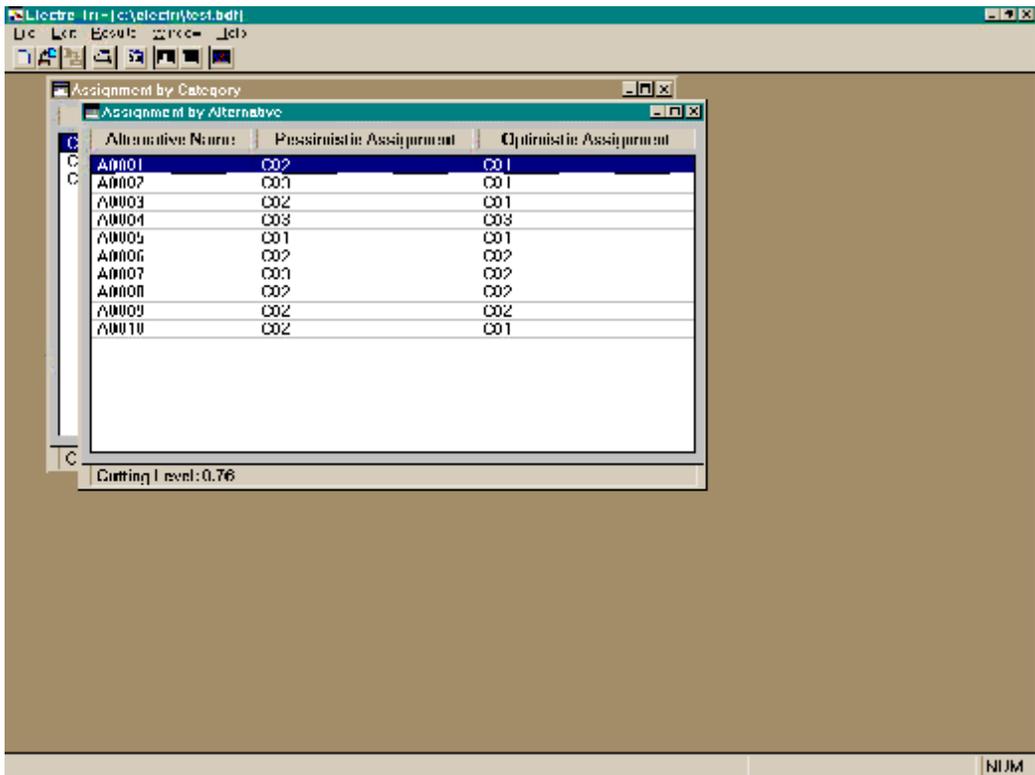
5.4.3 Intermediary results

So as to get insights from the results, it is possible to check the intermediary results. The ELECTRE TRI 2.0 software gives the possibility to :

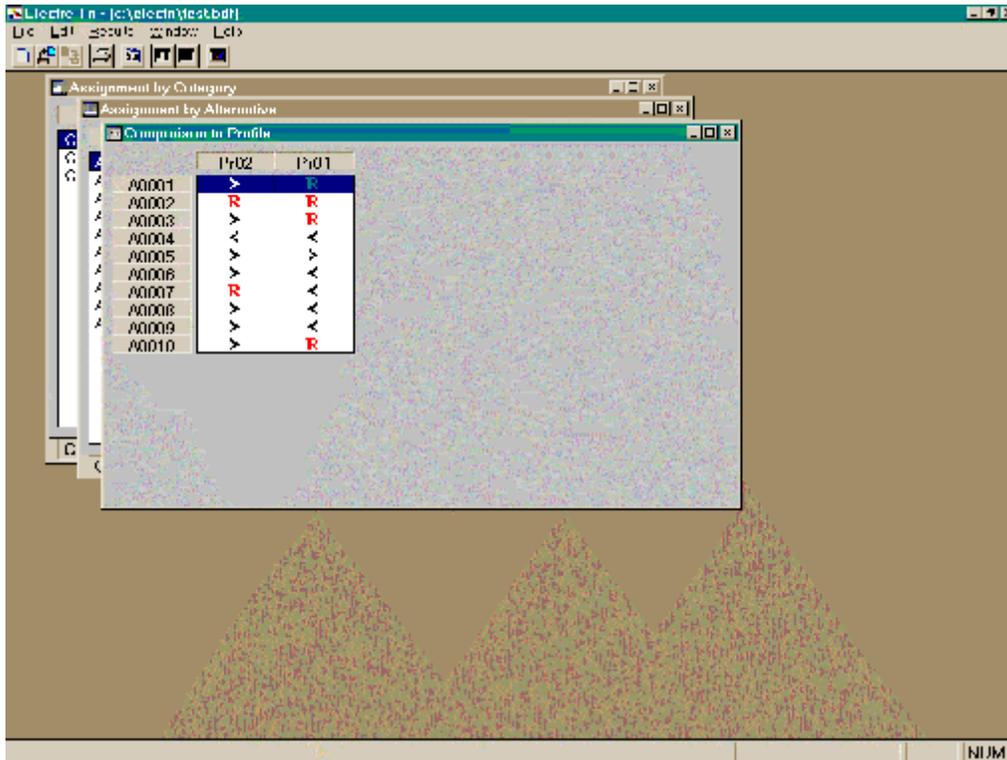
- ◆ check the comparisons of all alternatives to each limit profile. The symbols used are **f** (or **p**) for a preference, **I** for indifference and **R** for incomparability (see Screen 5.22). Select the **Comparison to Profiles** option in the **Result** command menu to obtain this information.
- ◆ Display degree of credibility of the outranking relation between each profile b_h and each alternative a . On Screen 5.23, each cell of the table contains two values $s(a, b_h)$ and $s(b_h, a)$. *The value at the top of the cell corresponds to $s(a, b_h)$, the one at the bottom of the cell corresponds to $s(b_h, a)$.* Select the **Degrees of credibility** option in the **Result** command menu to obtain this information.
- ◆ Get a visual representation of an alternative and the profiles. The representation (see Screen 5.24) enable to visualize one alternative at a time (in red) together with the profiles specifying the limits of the categories (in blue). Select the **Degrees of credibility** option in the **Result** command menu to obtain this representation.
- ◆ Obtain a statistical synthesis of the assignments (see Screen 5.25), i.e., the proportion of alternatives assigned to each category for both assignment rules (optimistic and pessimistic). Select the **Statistics of assignment** option in the **Result** command menu to obtain this information.



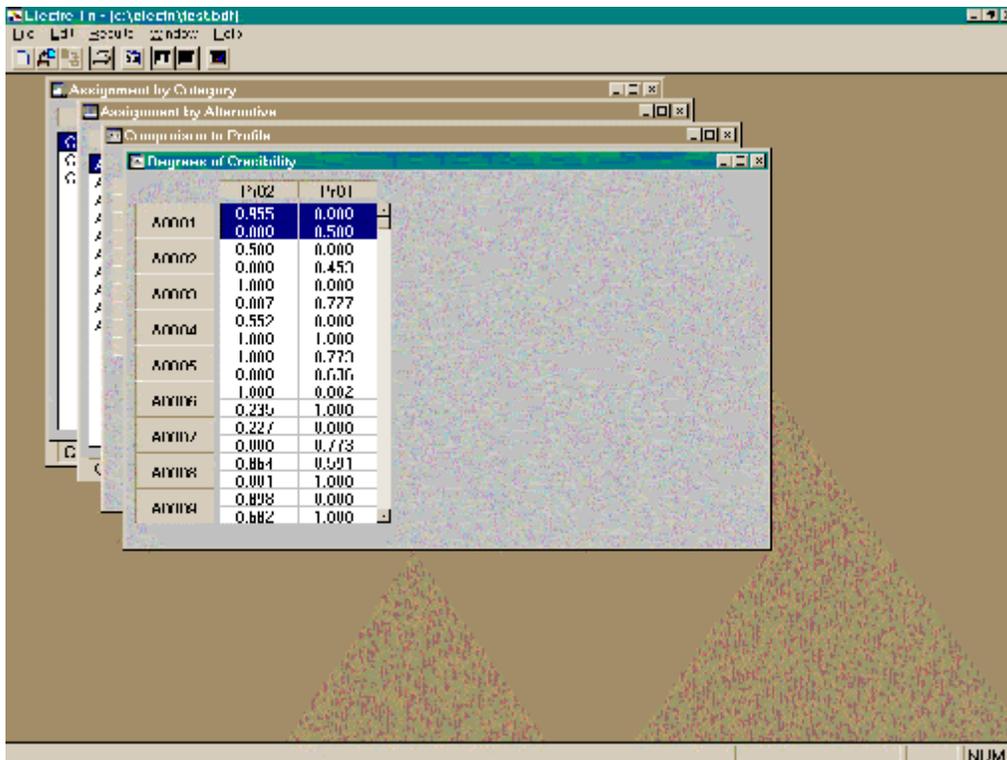
Screen 5.20: Assignment by category



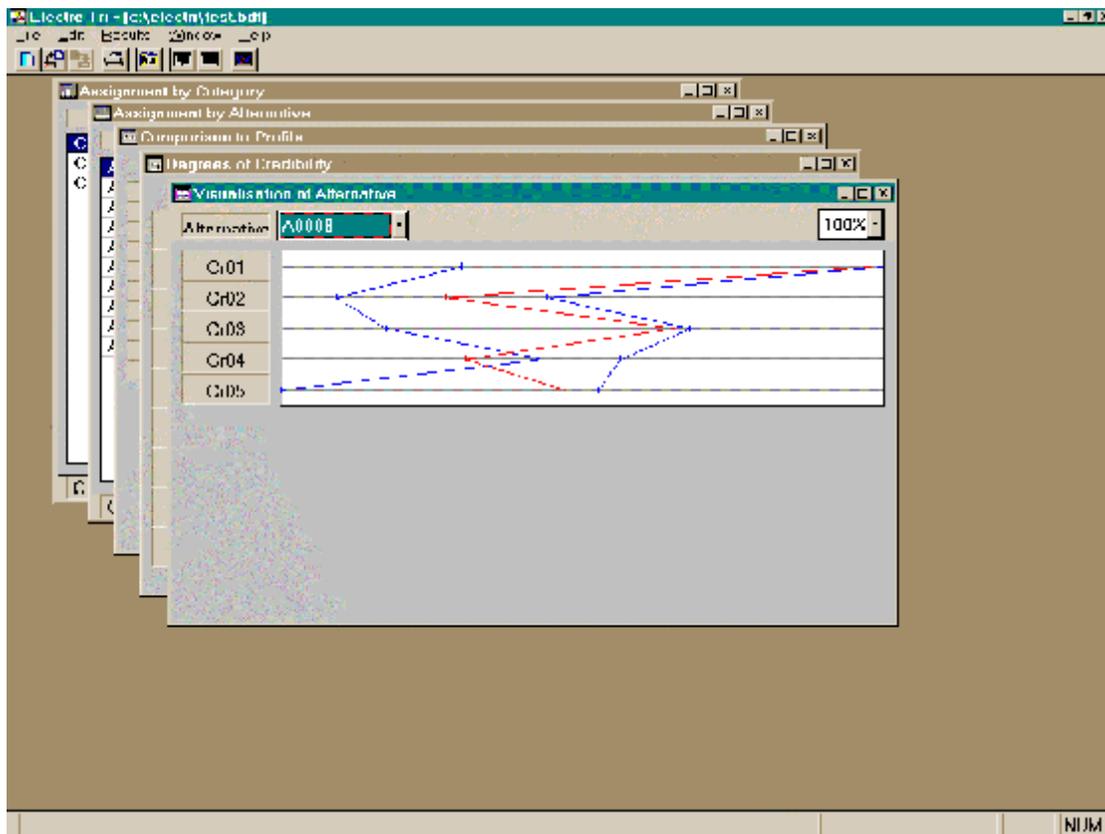
Screen 5.21: Assignment by alternative



Screen 5.22: Comparisons of alternatives to profiles



Screen 5.23: Degrees of credibility of the outranking relation



Category Name	Pessimistic Assignment	Optimistic Assignment
Cr01	10 % (1 of 10)	50 % (5 of 10)
Cr02	60 % (6 of 10)	40 % (4 of 10)
Cr03	30 % (3 of 10)	10 % (1 of 10)

Cutting Level: 0.76

Screen 5.25: Statistics of assignments

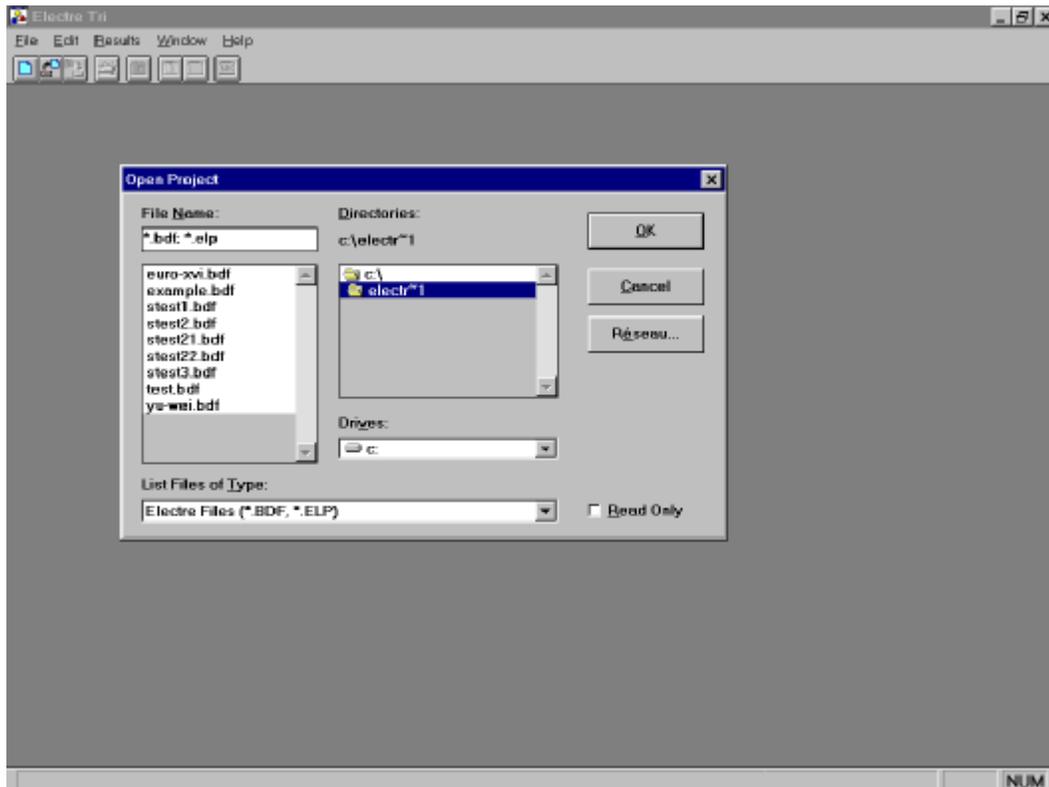
5.5 Closing the current project

When you want to finish a session with the project, choose the **C**lose Project option in the **F**ile command menu. This command will remove from memory the current data set. Remember that if you want to keep your dataset, you must use the **S**ave command before using this command.

5.6 Opening an existing project

The **O**pen Project option in the **F**ile command menu enables to load in memory a project created during a previous session of ELECTRE TRI and that has been saved on disk. The second button in the Tool bar  is a short-cut for this command.

You have to type the name of the project or to select it in the file list. You may choose the drive on which your file is saved and the directory in the **D**irectories dialog box. The window **F**iles gives a list of all files that have the mask proposed in the **F**ile Name dialog box. By default, ELECTRE gives a list of the files having the extension .BDF and .ELP in the current directory (see Screen 5.26).



Screen 5.26: opening an existing project

5.7 Loading data from the previous version of ELECTRE TRI

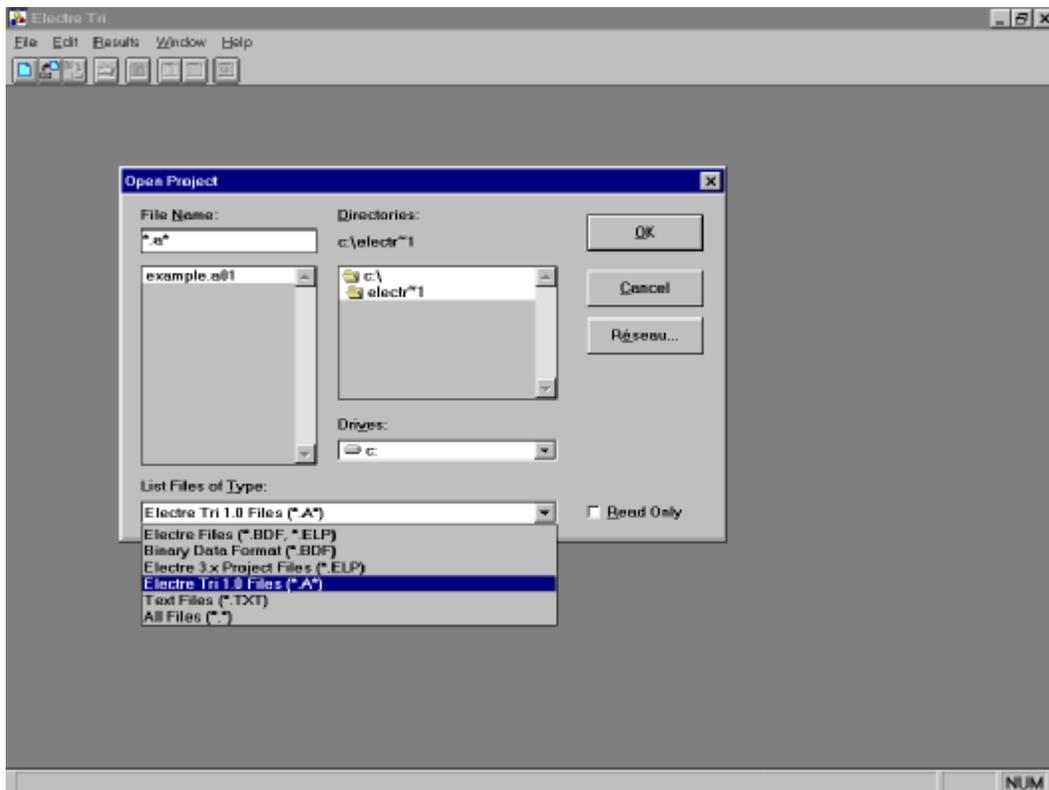
In the first version of the ELECTRE TRI software, a project consisted in three separate files:

1. a file with extension *.a* which contains the alternatives,
2. a file with extension *.k* which contains the definition of categories,
3. a file with extension *.c* which contains the definition of the criteria.

In the new ELECTRE TRI implementation, the whole project is saved in a single file (with extension .bdf, i.e., **binary data format**). However, the new implementation enables the user of the previous version to load projects saved with ELECTRE TRI 1.0. In order to do so, proceed as follows:

1. put the three files corresponding to the project (XXX.A01, XXX.K01 and XXX.C01) in the same directory,
2. select the **Open Project** command in the **Files** menu command,
3. select the *<Electre Tri 1.0 Files (*.A*)>* option in the *<List Files of Type>* combo box, see Screen 5.27.

4. select the directory and the file to be loaded; when you click on the  button, your file is loaded.



Screen 5.27: opening a project saved with the previous version of ELECTRE TRI

5.8 Loading a set of alternatives from Excel

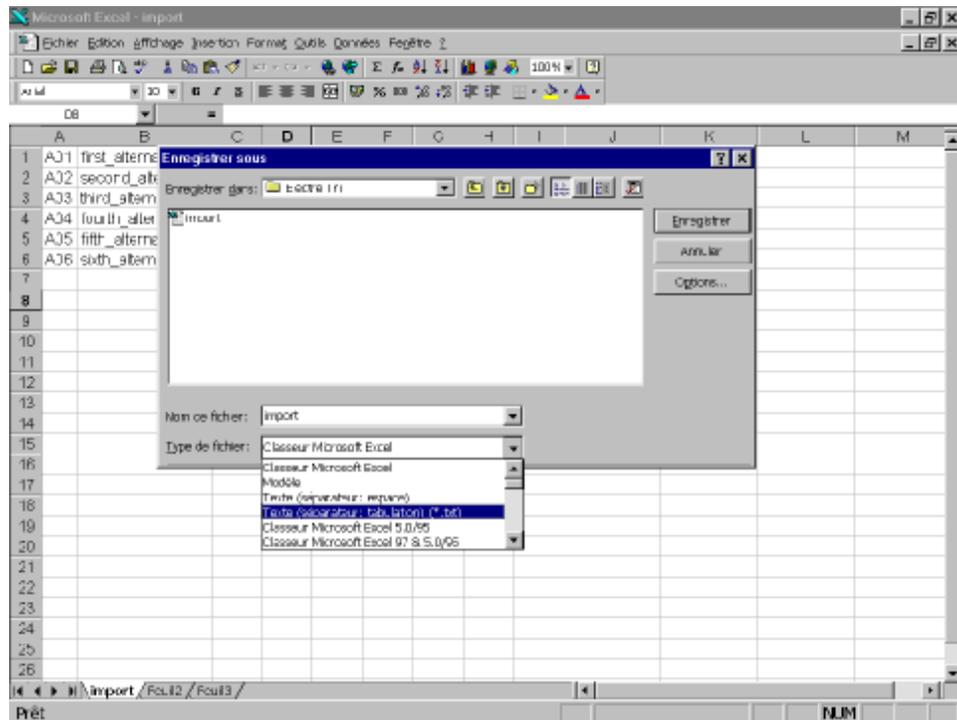
It is also possible to import a set of alternatives from Spreadsheets (such as Excel or any other spreadsheet that generates ASCII files). In order to do so, proceed as follows:

1. Structure your data (concerning the alternatives to be imported) in an Excel file using the following format (see Screen 5.28):
 - each line corresponds to an alternative,
 - the code of the alternative is specified in the first column (8 characters maximum without any spaces),
 - the name of the alternative is specified in the second column (255 characters maximum without any spaces),

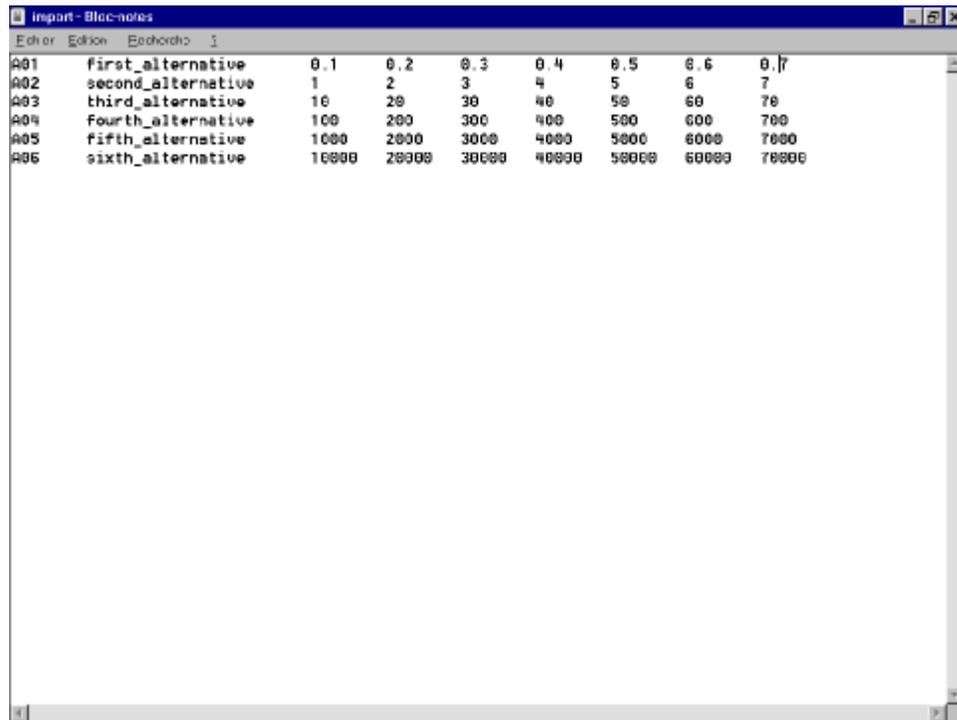
	A	B	C	D	E	F	G	H	I	J	K	L	M
1	A01	first_alternative	0,1	0,2	0,3	0,4	0,5	0,6	0,7				
2	A02	second_alternative	1	2	3	4	5	6	7				
3	A03	third_alternative	10	20	30	40	50	60	70				
4	A04	fourth_alternative	100	200	300	400	500	600	700				
5	A05	fifth_alternative	1000	2000	3000	4000	5000	6000	7000				
6	A06	sixth_alternative	10000	20000	30000	40000	50000	60000	70000				

Screen 5.28: Structure of the Excel file in order to import alternatives

2. make sure that the decimal separator is a "point" (.),
3. save your file (**Save As**) using the file type <Text (separator Tab) (*.txt)>, see Screen 5.29, the text file corresponding to the data in Screen 5.28 is shown in Screen 5.30.
4. select the **New Project** option in the **File** command menu.
5. select the **Dataset** option in the **Edit** command menu; insert as many criteria as necessary (7 in the case of the excel file shown in Screen 5.28), see §5.2.2,
6. select the **Import Alternatives** option in the **File** command menu, and choose your *.txt in the open file window; your alternatives are imported when you click on the **OK** button.



Screen 5.29: exporting an ascii file with Tab separators from Excel



Screen 5.30: Ascii file generated with Excel

5.9 An ELECTRE TRI Assistant session

So as to illustrate ELECTRE TRI Assistant possibilities, let us consider an hypothetical preference elicitation process. This process relies on the data from [Yu 92]. Let us suppose that our fictitious DM is able to elicit directly the profiles and thresholds as presented in Table 5.1 but has difficulties with expressing directly the importance coefficients.

In order to load this data load the file named `eta-ex.bdf` (see Screen 5.6).

Instead, our DM is able to express some assignment examples. These examples are reported in Table 5.2 and correspond to a subset $A^* \subseteq A$ of 15 alternatives hollistically assigned by the DM to a specific category (these examples were obtained by ELECTRE TRI pessimistic assignment rule using the weights from Table 5.1; the only exception is alternative a_5 for which the assignment has been voluntarily changed from C_2 to C_1 in order to check the influence of an "incorrect" assignment on the inference procedure.

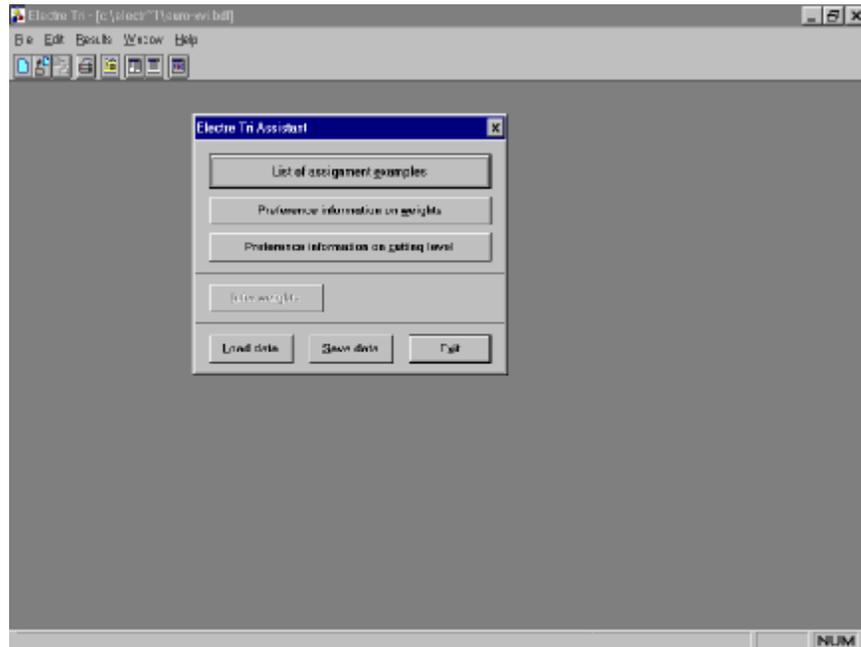
	g_1	g_2	g_3	g_4	g_5	g_6	g_7
k_j	3.0	1.5	1.5	1.5	1.5	1.5	2.0
$g_j(b_2)$	14.0	29.0	29.0	29.0	29.0	29.0	27.0
$q_j(b_2)$	0.64	1.56	1.56	1.56	1.56	1.56	2.04
$p_j(b_2)$	1.28	3.17	3.17	3.17	3.17	3.17	4.15
$v_j(b_2)$	1.92	9.51	9.51	9.51	9.51	9.51	16.60
$g_j(b_1)$	17.0	43.0	43.0	43.0	43.0	43.0	40.0
$q_j(b_1)$	0.67	1.84	1.84	1.84	1.84	1.84	2.56
$p_j(b_1)$	1.34	3.73	3.73	3.73	3.73	3.73	5.19
$v_j(b_1)$	2.01	11.19	11.19	11.19	11.19	11.19	20.76

Table 5.1: Preferential data in [Yu 92]

Name	g_1	g_2	g_3	g_4	g_5	g_6	g_7	Desired Category
a_{19}	13.02	15.74	18.02	7.24	79.21	42.63	79.32	C_1
a_{23}	13.66	11.01	14.11	70.55	69.01	18.77	42.39	C_1
a_{41}	13.04	7.99	22.44	7.24	31.40	14.83	58.65	C_1
a_{49}	13.48	1.05	18.02	6.45	31.40	18.77	100.00	C_1
a_{68}	9.91	7.99	14.11	7.24	12.92	3.02	58.65	C_1
a_5	14.43	11.01	18.02	29.25	22.16	31.73	39.44	C_1
a_{27}	13.51	14.02	18.02	29.25	22.16	8.91	39.44	C_2
a_{29}	13.39	11.01	18.02	17.36	22.16	8.91	39.44	C_2
a_{55}	12.14	7.99	18.02	5.46	22.16	3.02	39.44	C_2
a_{62}	11.07	7.99	18.02	5.46	12.92	3.02	39.44	C_2
a_{66}	10.25	7.99	14.11	15.58	12.92	8.91	20.24	C_3
a_{69}	10.65	11.01	10.47	3.69	3.76	8.91	20.24	C_3
a_{84}	8.26	7.99	10.47	3.69	3.76	3.02	20.24	C_3
a_{93}	1.96	6.74	15.58	3.69	3.76	3.02	20.24	C_3
a_{94}	3.86	1.96	3.02	3.69	3.76	3.02	20.24	C_3

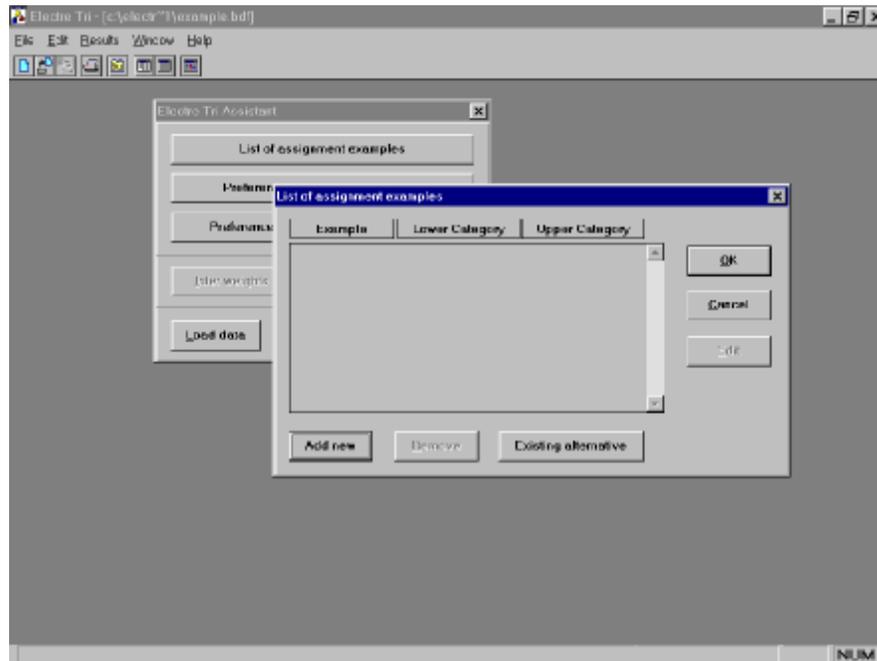
Table 5.2: Set of assignment examples

In order to enter Electre Tri Assistant, select the **Electre Tri Assistant** option in the **Edit** command menu. You obtain the Electre Tri assistant menu (see Screen 5.31).



Screen 5.31: Electre Tri Assistant menu

In order to enter the assignment examples click on the **List of assignment examples** button. You obtain the *List of assignment example* window wich displays an empty list (see Screen 5.32).



Screen 5.32: List of assignment example window

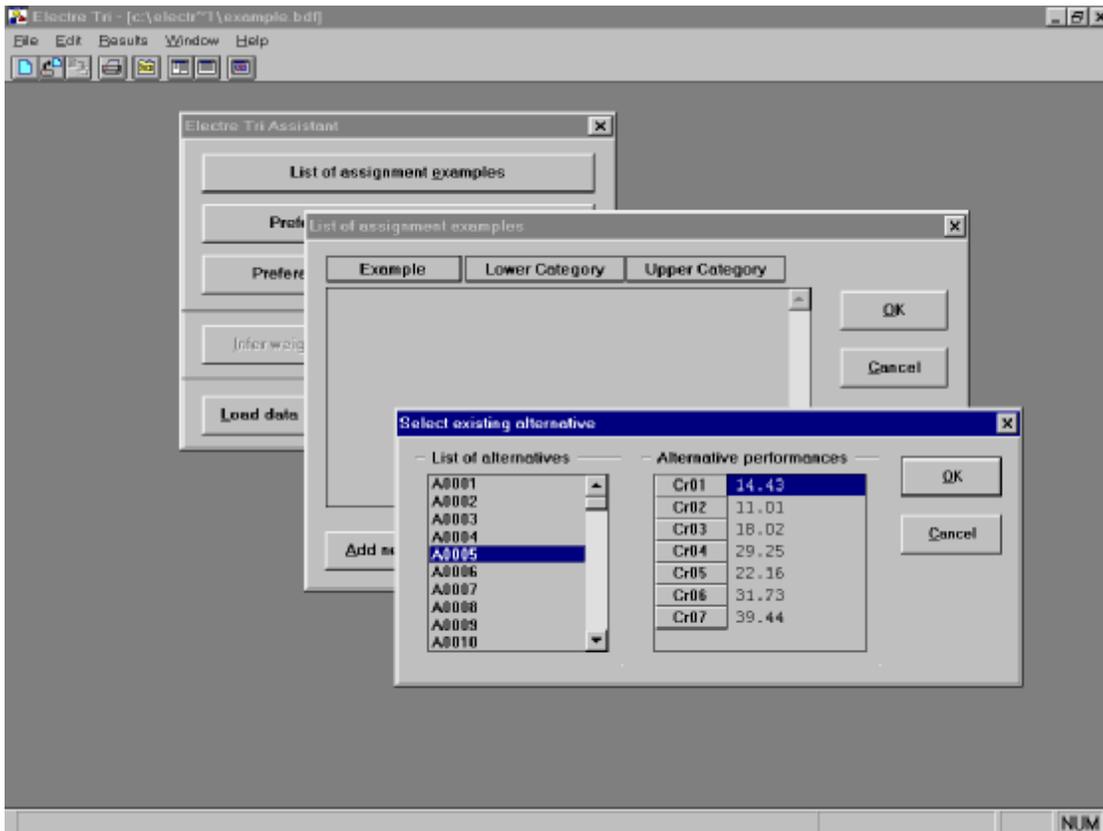
Through the *List of assignment example* window, it is possible to specify assignment example corresponding to alternatives for which the DM gives a holistic assignment. Such alternative can be:

- an existing alternative (i.e., one already input in the project), use the **Existing alternative** button;
- or a fictitious one designed for this creating an assignment example (use the **Add new** button).

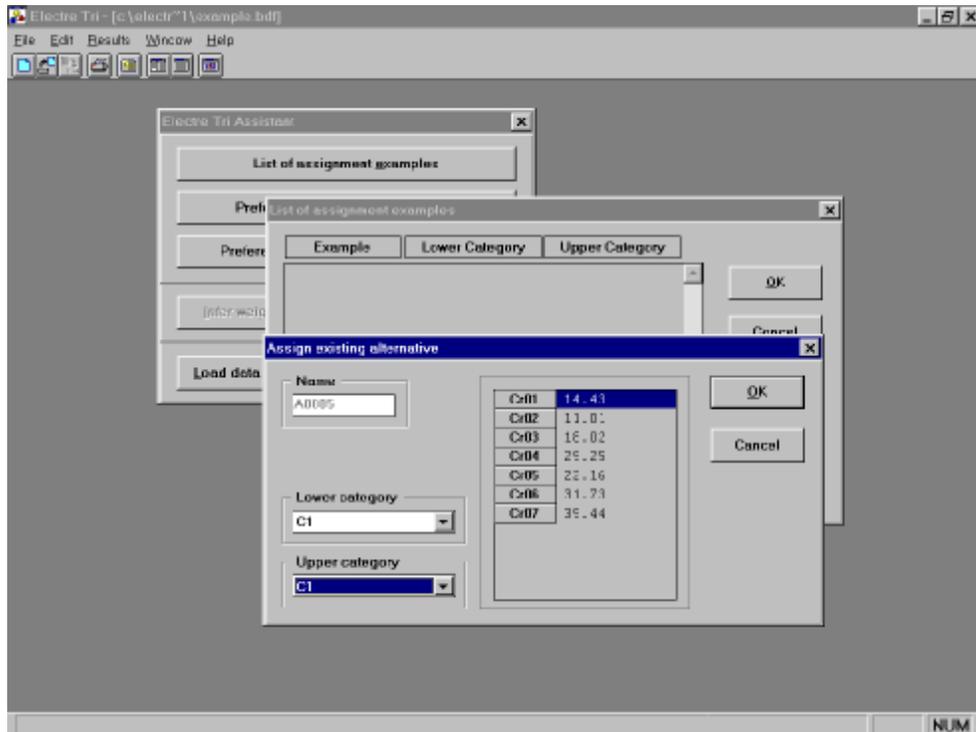
Imprecise assignments are accepted, i.e., the DM can express an hesitation in the assignment of an alternative *a* by specifying a subset of consecutive categories to which *a* could be assigned.

In order to add an assignment example corresponding to an existing alternative, use the **Existing alternative** button and proceed as follows :

1. select an alternative in the list of existing alternative (see Screen 5.33),
2. specify the minimum and maximum category to which the selected alternative should be assigned (see Screen 5.34).

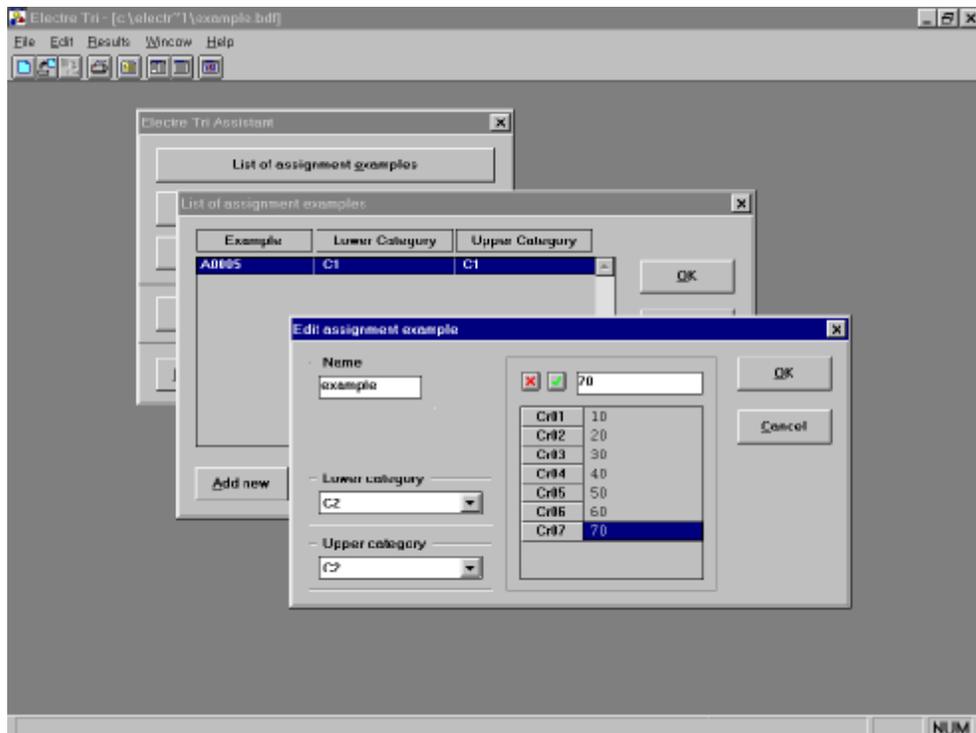


Screen 5.33: Select existing alternative window



Screen 5.34: Assign existing alternative window

In order to add an assignment example corresponding to an new alternative, use the **Add new** button. You must then input the name (see Screen 5.35), performances and desired category for this new alternative.

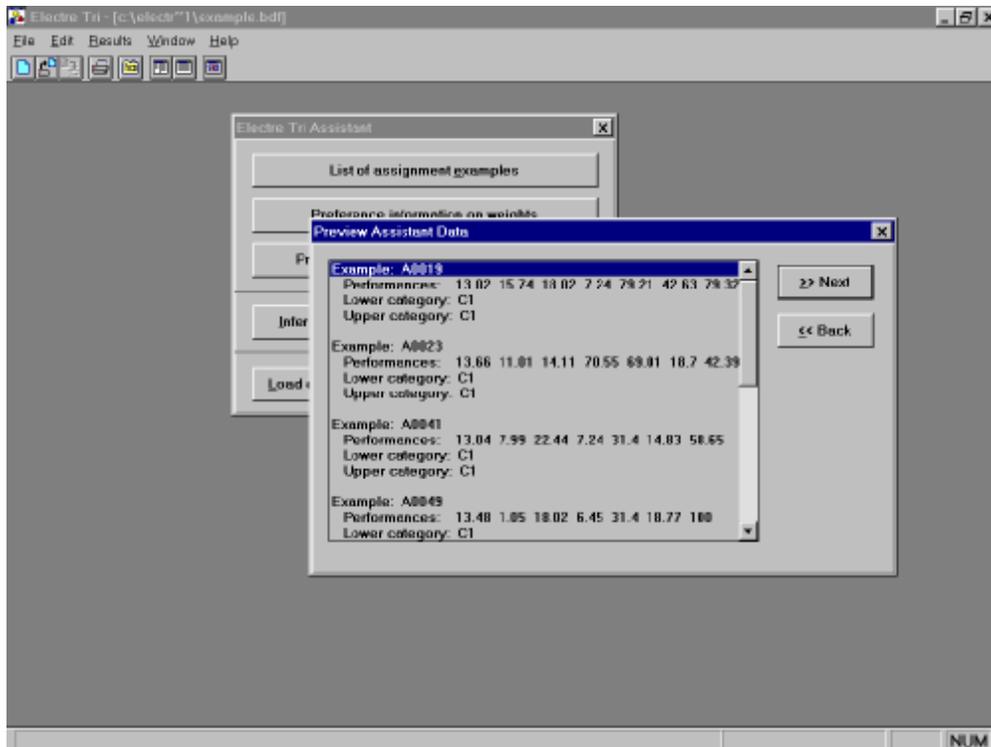


Screen 5.35: Edit assignment example window

When you have input all assignment examples, you can save this data using **Save data** button (see Screen 5.31). It will be possible to load this information in a future Electre Tri Assistant session using the **Load data** button. If you do not want to enter all assignment examples presented in Table 5.2, you may load them by loading the file named `eta-ex.eta`.

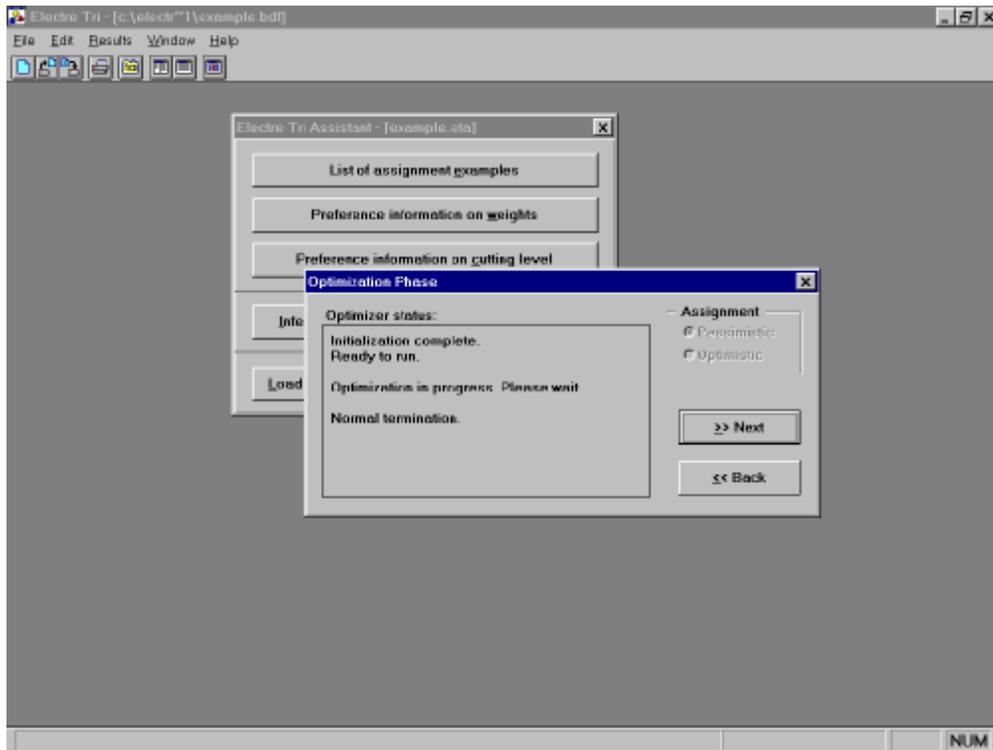
Let us suppose that the DM wants to get a first proposal for the weights using the assignment examples described in Table 5.2 without giving any additional information. In order to do so, click on the **Infer weights** button on the Electre Tri Assistant menu (see Screen 5.31).

Electre Tri Assistant will then display the information on the the basis of which the weights will be inferred (see Screen 5.36). From this screen, you may either change the data (click on the **<< Back** button) or continue in order to infer the weights (click on the **>> Next** button).

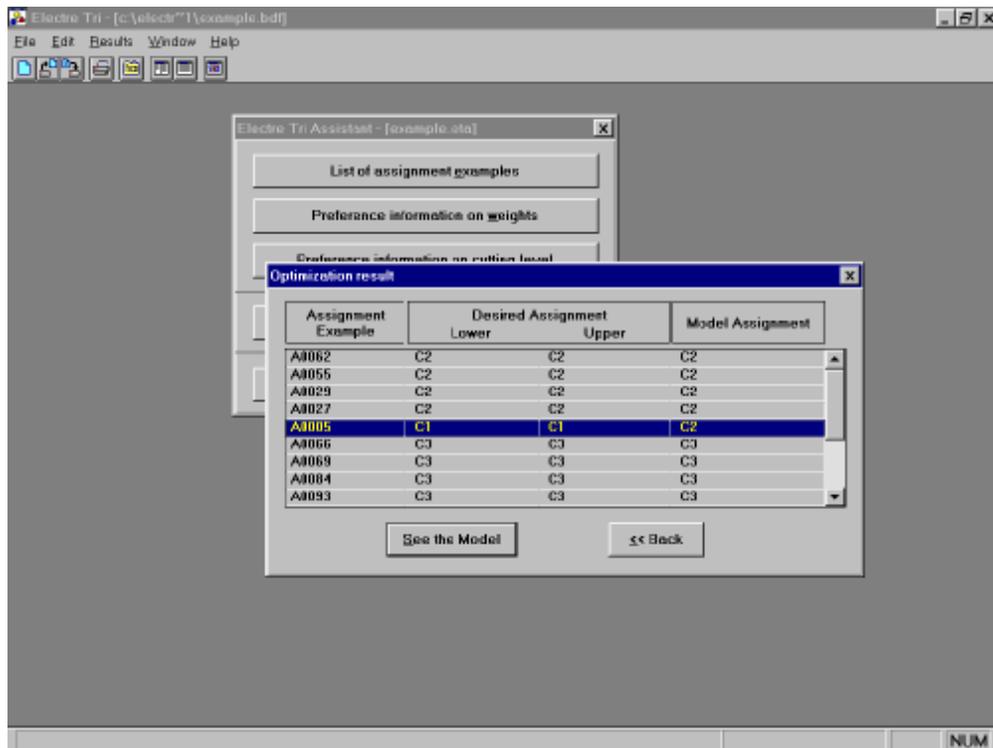


Screen 5.36: Preview assistant data window

After clicking on the **>> Next** button, the user must specify the Electre Tri assignment procedure on the basis of which the weight inference will be grounded (see Screen 5.37), i.e., either pessimistic or optimistic (in Electre Tri version 2.0a, it is only possible to infer the weights on the basis of the pessimistic assignment procedure).



Screen 5.37: Optimization phase window

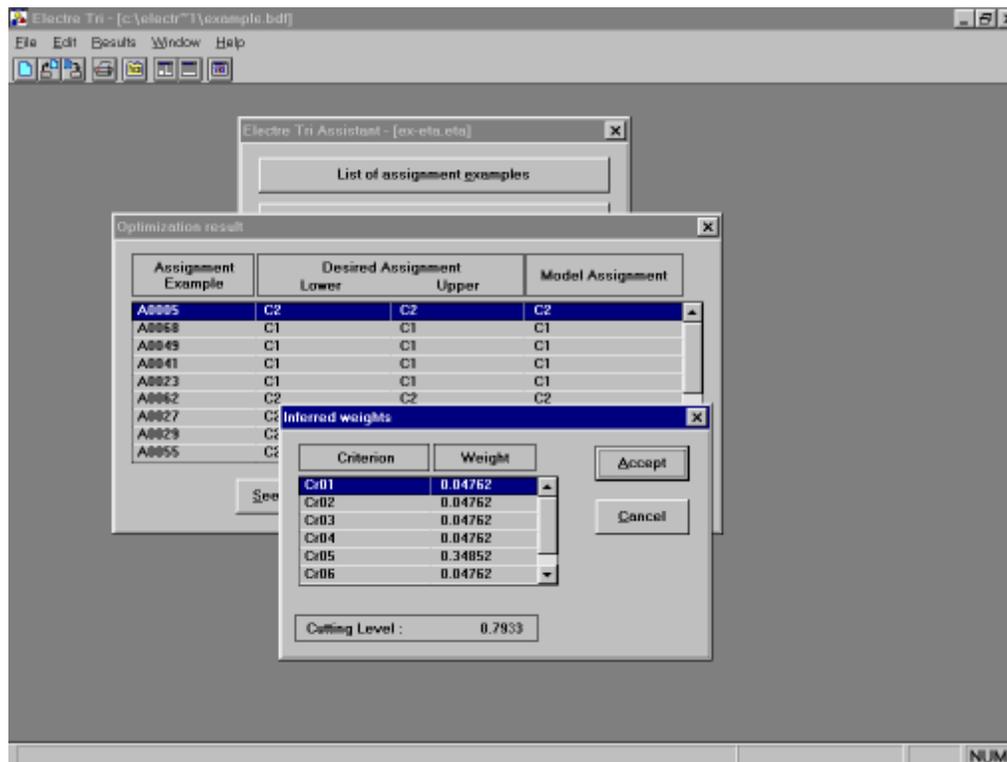


Screen 5.38: Optimization results window

The output of the computations (see Screen 5.38) shows that the inferred model is not able to assign all alternatives to their respective "desired" category; a_5 is assigned to C_2 instead of C_1 and appears in a different color.

Considering this first result, let us suppose that the DM revises his/her judgment concerning the assignment of alternative a_5 (by stating that a_5 should be assigned to C_2 in the Edit Assignment Examples screen, see Screen 5.34) and reruns the optimisation phase. After this second optimisation phase, all alternatives are assigned using the inferred weights consistently with the DM. In order to see the inferred weight vector (0.048, 0.048, 0.048, 0.048, 0.349, 0.048, 0.413)

and $\lambda=0.793$, click on the **See the Model** button; the weights and cutting level are displayed on the Screen 5.39.



Screen 5.39: Inferred weights window

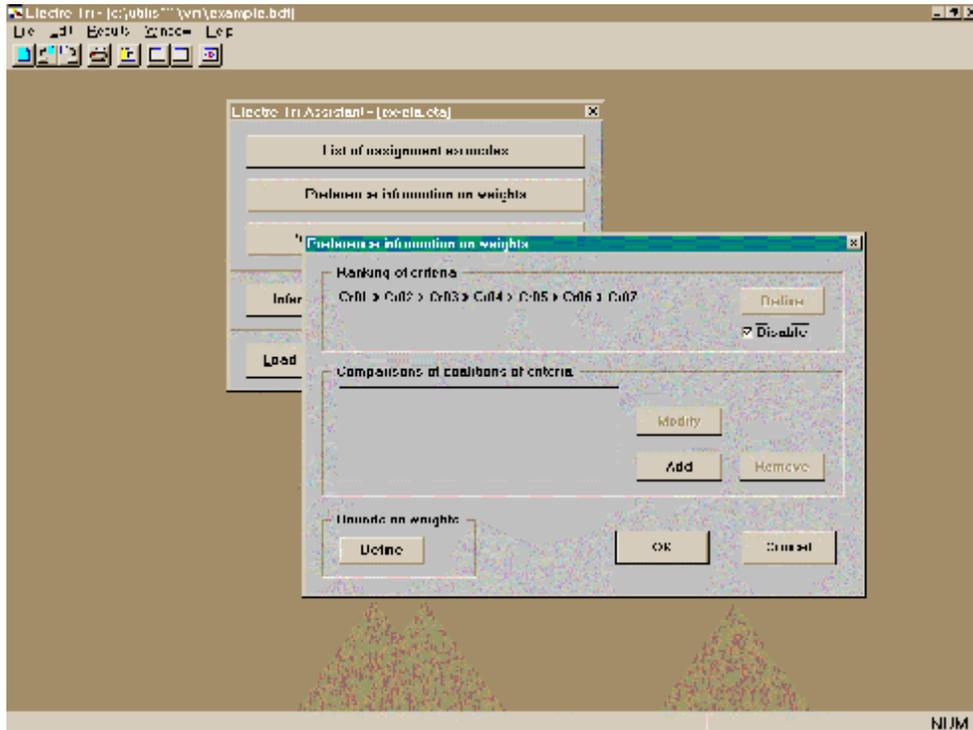
Our DM considers that the obtained weights do not express adequately his/her opinion concerning the importance of criteria. He/she is willing to add additional constraints by stating:

- a pre-order on criteria according to their relative importance
- comparisons of coalitions of criteria,

As our DM is surprised that the inferred weight of g_1 is low; he/she imposes g_1 to be more important than every other criterion, except g_7 , moreover he/she would like to add intuitive information concerning the ranking of criteria in terms of importance. In consequence, he/she specifies the following importance ranking on criteria:

$$g_7 \gg g_1 \gg g_5 \gg g_2 \approx g_4 \gg g_3 \approx g_6, \text{ i.e., } k_7 > k_1 > k_5 > k_2 = k_4 > k_3 = k_6$$

In order to do so, click on the **Preference information on weights** button. This leads you to the Screen 5.40. In order to specify a ranking of the criteria considering their relative importance, click on the check-box **Disable** (default value is on) and then on the **Define** button (see Screen 5.41).



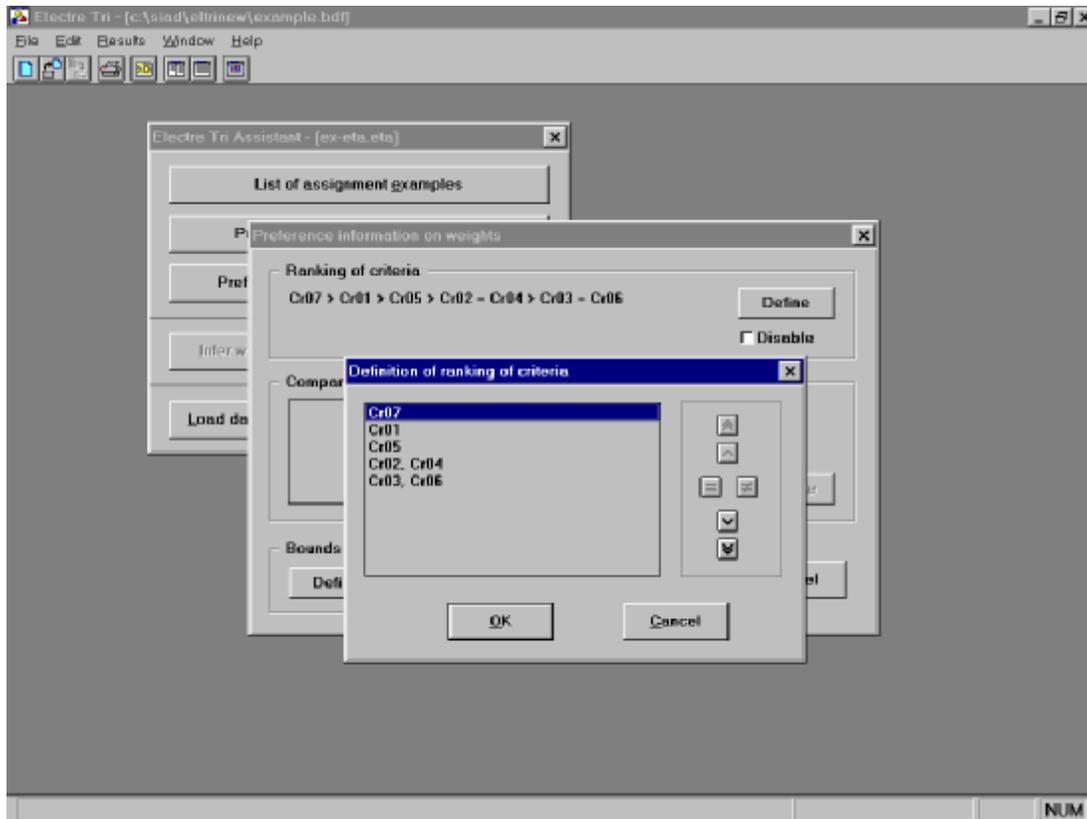
Screen 5.40: Preference information of weights screen

The window shown in Screen 5.41 is composed of two parts. The left part displays the ranking of the criteria while the right part, provides buttons that enables the user to modify the ranking. In order to specify a modification in the ranking, it is necessary to proceed as follows:

1. select a criterion (or range of criteria),
2. click on a button on the right part of the screen.

The meaning of the buttons are the following :

-  places the selected criterion in the first place in the preorder,
-  places the selected criterion in the last place in the preorder,
-  places the selected criterion one place above in the preorder,
-  places the selected criterion one place below in the preorder,
-  places the selected criteria at the same rank in the preorder,
-  splits the selected equally important criteria in the preorder,



Screen 5.41: definition of ranking of criteria screen

The initial default ranking is $g_1 \gg g_2 \gg g_3 \gg g_4 \gg g_5 \gg g_6 \gg g_7$. In order to specify the ranking $g_7 \gg g_1 \gg g_5 \gg g_2 \approx g_4 \gg g_3 \approx g_6$, the following steps are necessary:

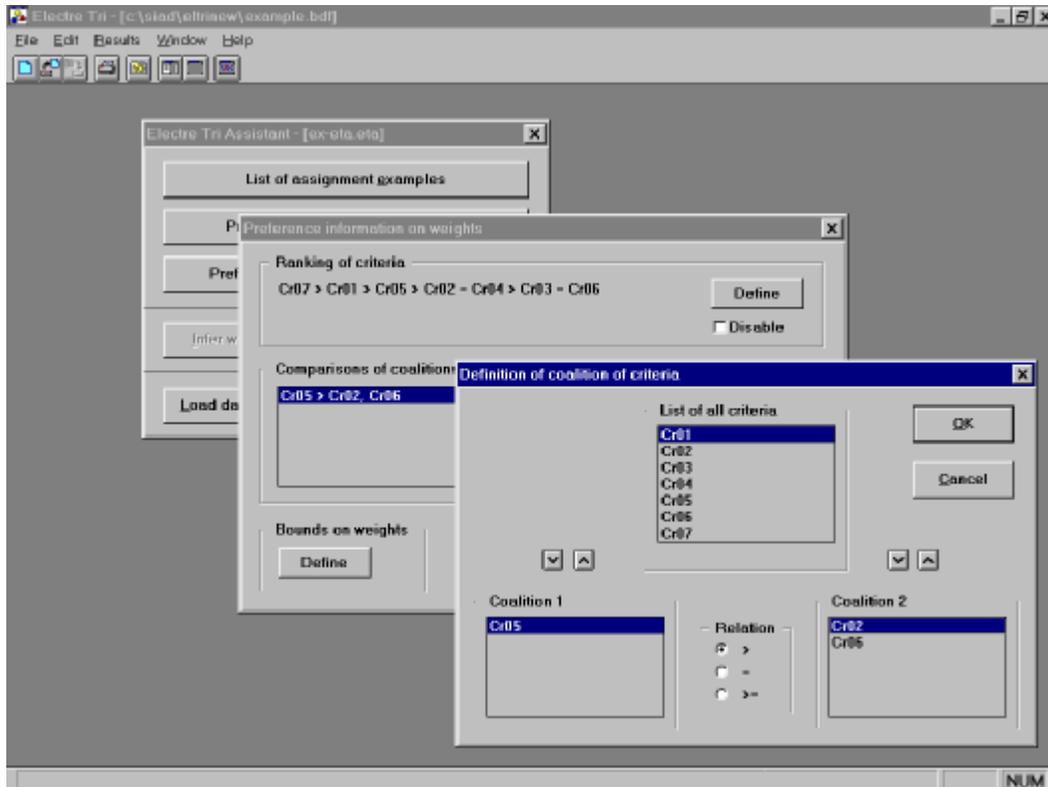
1. select g_7 and click on
2. select g_5 and click three times on
3. select g_4 and click once on
4. select g_2 and g_4 and click on
5. select g_3 and g_6 and click on

Further, the DM considers that a preference on criterion g_5 is more important than a conjoint preference on g_2 and g_6 . This information can be expressed through a comparison of coalitions of criteria in term of importance:

$$g_5 \gg \{g_2, g_6\} \text{ , i.e., } k_5 > k_2 + k_6$$

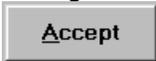
In order to add such information, click on the button in the middle of the window “preference information on weights” (see Screen 5.40). Specifying a comparison of two coalitions of criteria proceeds as follows (see Screen 5.42).

1. specify coalition 1 by selecting the criteria one by one in the list of all criteria and clicking on the button  above the list of coalition 1
2. specify coalition 2 by selecting the criteria one by one in the list of all criteria and clicking on the button  above the list of coalition 2
3. select the comparison mode, = (equally important), > more important or \geq (at least as important)



Screen 5.42: definition of a comparison of coalitions of criteria screen

The optimisation phase taking into account additional preference information is then performed again. In result of optimisation, the inferred weights still assign all alternatives consistently with the DM. The inferred weight vector (0.246, 0.058, 0.048, 0.058, 0.236, 0.048, 0.307) and $\lambda=0.847$ are displayed on a screen similar to the one in Screen . As the DM considers these weights to reflect adequately his/her opinion concerning the importance of criteria, he/she may

takes this vector into account (by clicking on the  button) so as to assign all other alternatives using these weights.

Let us remark that, in general, the weights resulting from the optimisation phase are not unique. In order to learn about the interval of variation of weights giving the same value for the objective function, one should perform a standard sensitivity analysis. However, the information stemming from such analysis may not be as useful as the insights that the user can get on his/her preferences through a trial and error process.

7. References

G. Anandalingam and C. E. Olsson, 1989, "A Multi-stage Multi-attribute Decision Model for Project Selection", *EJOR*, volume 43, pages 271-283.

A. Andenmatten, 1995, "Evaluation du risque de défaillance des émetteurs d'obligations, une approche par l'aide multicritère à la décision", Presses Polytechniques et Universitaires Romandes.

C. Bana e Costa, 1996, "Les problématiques de l'aide à la décision : vers l'enrichissement de la trilogie Choix-Tri-Rangement", *RAIRO/Operations Research*, volume 30, number 2, pages 191-216.

C. Bana e Costa, 1992, "Absolute and Relative Evaluation Problematiques: The Concept of Neutral Level and the MCDA Robot Technique", in "Multicriteria Decision Making: Methods, Algorithms and Applications", M. Cerny and D. Gluckaufova and D. Loula (eds.), Czechslovak Academy of Sciences, Prague.

A. I. Dimitras, C. Zopounidis and C. Hurson, 1995, "A Multicriteria Decision Aid Method for the Assessment of Business Failure Risk", *Foundation of Computing and Decision Sciences*, volume 20, number 2, pages 99-112.

V. Gabrel, 1994, "Méthodologie pour la planification de production de système d'observation de la terre par satellites - Aspects algorithmiques et multicritères", Phd dissertation, Université Paris-Dauphine, France.

S. Greco and B. Matarazzo and R. Slowinski, 1998a, A new rough set approach to multicriteria and multiattribute classification", in L. Polkowski and A. Skowron (eds.), "Rough Sets and Current Trends in Computing", pages 60-67, Springer-Verlag, Berlin.

S. Greco and B. Matarazzo and R. Slowinski, 1998b, A new rough set approach to evaluation of bankruptcy risk, in C. Zopounidis (ed.) "Operational tools in the management of financial risks", pages 121-136, Kluwer ", Dodrecht.

Y. Groleau, N. Khoury and J.-M. Martel, 1995, "procédure multicritère pour l'octroi de crédit commercial", *Rendu du ASAC Management Science / Recherche Opérationelle*, volume 16, number 2, pages 39-47.

E. Jacquet-Lagrèze and J. Siskos, 1982, "Assessing a set of additive utility functions for multicriteria decision-making, the UTA method", *EJOR*, volume 10, number 2, pages 151-164.

E. Jacquet-Lagrèze and R. Mezzani and R. Slowinski, 1987, "MOLP with an interactive assessment of a piecewise-linear utility function", *EJOR*, volume 31, number 3, pages 350-357.

E. Jacquet-Lagrèze, 1990, Interactive assessment of preferences using holistic judgements: the PREFCALC system", in "Readings in Multiple Criteria Decision Aid", Bana e Costa C. (ed.), Springer-Verlag, pages 335-350.

O. I. Larichev and H. M. Moshkovich, 1994, "An Approach to Ordinal Classification Problems", International Transactions in Operational Research, volume 1, number 3, pages 375-385.

O. I. Larichev, H. M. Moshkovich and E. M. Furems, 1988, "Decision Support System CLASS", in "New Directions in Research on Decision Making", B. Brehmer, H. Jungermann, P. Lourens and G. Sevon (eds.), Elsevier Science, North Holland, pages 303-315.

R. Massaglia and A. Ostanello, 1991, "N-Tomic: A Support System for Multicriteria Segmentation Problems", in "Multiple Criteria Decision Support", P. Korhonen, A. Lewandowski and J. Walenius (eds.), Lecture Notes in Economics and Mathematical Systems, volume 356, IASA, Proceedings of the International Workshop, Helsinki, pages 167-174.

J. Moscarola and B. Roy, 1977, "Procédure automatique d'examen de dossiers fondée sur une segmentation trichotomique en présence de critères multiple", RAIRO / Operations Research, volume 11, number 2, pages 145-173.

V. Mousseau and R. Slowinski, 1998, "Inferring an ELECTRE TRI Model from Assignment Examples", Journal of Global Optimization, volume 12, pages 157-174.

V. Mousseau, J. Figueira and J.Ph. Naux, 1997, "Using assignment examples to infer weights for ELECTRE TRI method: Some experimental results", cahiers du LAMSADE n° 150, Université Paris-Dauphine.

V. Mousseau, 1995, "Eliciting information concerning the relative importance of criteria", in "Advances in Multicriteria Analysis", Pardalos P., Siskos Y. and Zopounidis C.(eds.), pages 17-43, Nonconvex Optimization and its Applications, Kluwer Academic Publishers, Dordrecht.

V. Mousseau, 1993, "Problèmes liés à l'évaluation de l'importance relative des critères en aide multicritère à la décision : Réflexions théoriques, expérimentations et implémentations informatiques", PhD dissertation, Université Paris-Dauphine.

R. Nadeau and J.-M. Martel and L. Kiss, 1991, "ELECCALC: Modelling the decision maker's preferences with ELECTRE II", in "Proceeding of the international workshop on Multicriteria Decision Making: Methods, Algorithm & Applications", Cerny M., Gluckofova D. & Loula D.(eds.), Liblice.

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

P. Perny, 1998, Multicriteria Filtering Methods based on Concordance and Non-Discordance Principles, Annals of Operations Research, volume 80, pages 137-165.

Z. Pawlak and R. Slowinski, 1994, "Rough set approach to multi-attribute decision analysis", EJOR, volume 72, pages 443-459.

B. Roy, 1981, "A Multicriteria Analysis for Trichotomic Segmentation Problems", in "Multiple Criteria Analysis: Operational Methods", P. Nijkamp and J. Spronk (eds.), Gower Publishing Company, Aldershot, England, pages 245-257.

B. Roy, 1989, "Main sources of inaccurate determination, uncertainty and imprecision in decision models", *Mathematical and Computer Modelling*, volume 12, number 10-11, pages 1245-1254.

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

B. Roy and Ph. Vincke, 1984, "Relational systems of preferences with one or more pseudo-criteria: some new concepts and results", *Management Science*, volume 30, number 11, pages 1323-1334.

Y. Siskos and D. Yanacopoulos, 1985, "UTA STAR - an ordinal regression method for building additive value functions", *Investigacao Operational*, volume 5, pages 39-53.

R. Slowinski, 1991, "Interactive multiobjective optimization based on ordinal regression", in "Multiobjective Problems of Mathematical Programming" Lewandowski A. & Volkovich V. (eds.), volume 351, LNEMS, Berlin, Springer-Verlag, pages 93-100.

R. Slowinski, 1992, "Intelligent Decision Support: Handbook of Applications and Advances of the Rough Set Theory", Kluwer Academic Publishers, Dordrecht.

K. Slowinski, 1992, "Rough Classification of HSV Patients", in "Intelligent Decision Support: Handbook of Applications and Advances of the Rough Set Theory", Slowinski R. (ed.), Kluwer Academic Publishers, Dordrecht.

H. Tanaka, H. Ishibuchi and T. Shigenaga, 1992, "Fuzzy Inference System Based on Rough Sets and its Application to Medical Diagnosis", in "Intelligent Decision Support: Handbook of Applications and Advances of the Rough Set Theory", Slowinski R. (ed.), Kluwer Academic Publishers, Dordrecht.

M. Veilleux, N. Khoury and J.-M. Martel, 1996, "Système multicritère d'octroi de subventions visant le développement économique régional", Document de travail numéro 96-21, Université Laval.

D. Yannacopoulos, 1985, "Mise en place et expérimentation d'un SIAD multicritère : le système MINORA", PhD dissertation, Université Paris-Dauphine.

W. Yu, 1992a, "ELECTRE TRI : Aspects méthodologiques et manuel d'utilisation", document du LAMSADE n°74, Université Paris-Dauphine.

W. Yu, 1992b, "Aide multicritère à la décision dans le cadre de la problématique du tri : concepts, méthodes et applications", PhD dissertation, Université Paris-Dauphine.