

THESIS / THÈSE

MASTER EN SCIENCES INFORMATIQUES

Mise au point d'un système conversationnel pour l'aide à la décision multicritère

Mary, Luc

Award date:
1983

Awarding institution:
Université de Namur

[Link to publication](#)

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal ?

Take down policy

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

FACULTES UNIVERSITAIRES NOTRE-DAME DE LA PAIX (Namur).

MISE AU POINT D'UN SYSTEME
CONVERSATIONNEL POUR L'AIDE
A LA DECISION MULTICRITERE.

Mémoire présenté par
Luc MARY
en vue de l'obtention du
grade de Licencié et Maître
en Informatique.

Année académique 1982-1983.

Nous remercions Monsieur FICHEFET d'avoir
accepté la direction de ce mémoire.

Nous remercions également les membres de
l'Institut et du Centre de Calcul, qui de
près ou de loin, ont contribué à l'élabo-
ration de ce travail.

Nous remercions finalement tous ceux et
celles qui, de près ou de loin, ont contri-
bué à la réalisation de ce travail et de
ces études.

TABLE DES MATIERES.

	<u>pages</u> :
Introduction.	1
<u>Chapitre I</u> : Les contraintes imposées au logiciel d'aide à la décision multicritère.	3
I.1 Les contraintes associées au caractère multi- critère du processus de décision.	3
I.1.1 La nature du problème de décision.	3
I.1.2 La nature du processus de décision.	4
I.1.3 La nature de l'objectif à atteindre dans le processus d'agrégation.	6
I.1.4 Les types de processus d'agrégation.	7
I.1.4.1 La construction d'une fonction d'utilité.	7
I.1.4.2 L'établissement d'une relation de surclassement.	7
I.1.4.3 Optimisation en analyse ordinale des données.	8
I.2 Les contraintes associées à la nature des méthodes et à leur utilisation par l'homme.	10

pages :

I.2.1 La nécessité d'établir une interface entre l'utilisateur et les méthodes.	10
I.2.2 La nécessité de pouvoir réaliser des analyses de sensibilité .	12
I.2.3 La nécessité de fournir à l'utilisateur des résultats clairs.	12

Chapitre II : Des outils permettant une interface
souple entre l'utilisateur et les méthodes mul-
ticritères.

II.1 Les outils pouvant être utilisés pour l'éta- blissement des trois rapports de préférence.	14
II.1.1 La construction d'une relation de préfé- rence quelconque.	14
II.1.1.1 Principe.	14
II.1.1.2 Le schéma d'utilisation.	15
II.1.1.3 Les méthodes associées à certaines étapes d'utilisation.	15
II.1.1.3.1 La construction de la relation binaire.	15
II.1.1.3.2 Le test du type de relation et l'approximation d'une relation quelconque.	15

pages :

II.1.1.3.2.1 L'algorithme de test.	18
II.1.1.3.2.2 La méthode d'approximation d'une relation quelconque par un quasi-ordre ou d'un quasi-ordre par un préordre total.	19
II.1.1.4 Les limites de la construction d'une relation binaire.	22
II.1.2 L'utilisation de la méthode hiérarchi- que des comparaisons par paires de Saaty.	22
II.1.2.1 Rappel concernant la théorie des matrices.	22
II.1.2.2 Objectif.	24
II.1.2.3 Principe.	24
II.1.2.4 Schéma d'utilisation.	27
II.1.2.5 Explicitation du choix d'une échelle de rapport et de la construction de P.	27
II.1.2.6 Les limites de la méthode.	28
II.2 L'utilisation de méthodes multicritères explicatives pour déterminer les rapports de préférence entre critères.	29

II.3 L'utilisation de fonctions d'utilité pour déterminer les évaluations d'actions pour un critère.	30
II.3.1 Principe .	30
II.3.2 Condition d'utilisation de la technique du point de valeur médiane.	30
II.3.3 Principe de la technique du point de la valeur médiane.	31
II.3.4 Schéma d'utilisation.	31
II.3.5 Explicitation du test de cohérence.	32

Chapitre III: Les méthodes multicritères.

III.1 Définitions préliminaires.	34
III.2 La méthode Oreste.	36
III.2.1 Objectif de la méthode.	36
III.2.2 Explicitation de ces étapes.	36
III.2.2.1 La phase de projection sur D.	37
III.2.2.2 La phase de rangement.	38
III.2.2.3 La phase d'agrégation.	38
III.3 Des méthodes explicatives d'une préférence globale : les méthodes UTA.	42

pages :

III.3.1 Objectif de la méthode.	42
III.3.2 La construction de la fonction d'utilité pour la méthode UTA I.	43
III.3.2.1 Condition d'utilisation de UTA I.	43
III.3.2.2 La méthode de construction de $u_i(g_i)$.	44
III.3.2.3 Les hypothèses faites sur les utilités partielles $u_i(g_i)$.	44
III.3.2.4 La détermination de la fonction d'utilité $U(g)$.	45
III.3.3 La construction de la fonction d'uti- lité pour la méthode UTA II.	48
III.3.4 L'analyse post-optimale.	51
III.3.4.1 Justification de cette étape.	51
III.3.4.2 Principe de l'analyse optimale.	52
III.3.4.3 Application aux méthodes UTA.	53
III.3.5 Etablissement d'une relation de sur- classement.	54

pages :

III.4 Les méthodes Electre.	56
III.4.1 Le principe de construction des relations de surclassement.	57
III.4.2 La méthode Electre I.	58
III.4.2.1 Principe de la construction de la relation de surclassement.	58
III.4.2.2 La problématique de la sélection des bonnes actions".	59
III.4.2.2.1 Définition .	60
III.4.2.2.2 Problème soulevé par l'utilisation de quasi-noyaux.	61
III.4.2.2.3 Méthodes pour rechercher les quasi-noyaux.	61
III.4.2.2.3.1 Définitions préliminaires.	62
III.4.2.2.3.2 Principe de l'algorithme.	62
III.4.2.2.3.3 L'algorithme.	63
III.4.2.2.3.4 Explicitation des étapes de l'algorithme.	65
III.4.3 La méthode Electre II.	70

pages :

III.4.3.1 Objectif.	70
III.4.3.2 Principe de construction des relations de surclassement.	70
III.4.3.3 Principe de la détermination des classements.	71
III.4.3.4 Comparaison des 2 préordres obtenus.	72

Chapitre IV : L'implémentation du logiciel.

IV.1 Exécution.	75
IV.1.1 La forme du menu-exécution.	76
IV.1.2 La forme des traitements.	78
IV.1.2.1 Le squelette de la routine principale associée aux méthodes d'agrégation.	81
IV.1.2.2 Le squelette de la routine principale associée aux méthodes UTA.	83
IV.1.2.3 Description de la routine d'exécution des méthodes.	84
IV.1.2.3.1 Schéma de la routine d'exécution associée aux méthodes d'agrégation.	85
IV.1.2.3.2 Schéma de la routine d'exécution associée aux méthodes UTA.	86

pages :

IV.1.2.3.3 Précisions sur l'enregistrement des données et des résultats.	88
IV.2 Impression.	90
IV.3 Suppression.	92
 <u>Chapitre V : Expériences pratiques.</u>	
V.1 Le problème envisagé : l'identification des entérobactéries.	93
V.1.1 Les expérimentations réalisées.	96
V.1.1.1 Les expérimentations vis-à-vis de la méthode Electre III.	97
V.1.1.2 Les expérimentations vis-à-vis de la méthode Oreste.	100
Conclusion.	103
Bibliographie.	105

INTRODUCTION.

"Notre époque est bien marquée par le problème du choix : nous passerons dans l'avenir moins de notre temps à travailler et davantage à essayer de choisir".

Ces paroles de Louis Leprince-Ringuet expliquent peut-être l'effort considérable consacré depuis une bonne dizaine d'années aux problèmes multicritères.

Le fait, en effet, de confronter simultanément plusieurs critères dans un problème peut dépasser les capacités d'appréhension globale d'un individu. Tantôt, le décideur éprouve des difficultés à créer l'information nécessaire à la décision; d'une part, la saisie de toutes les interactions entre les actions ou entre les critères qui soit s'opposent, soit sont indépendants, soit sont dépendants à des degrés divers est très difficile; d'autre part, la traduction de ces interactions complexes sous des formes simples mais complètes n'est pas chose aisée. Tantôt, le décideur maîtrise difficilement une information qui, une fois complète, devient trop abondante que pour qu'il puisse en retirer une solution.

Ces difficultés ont imposé la nécessité de se doter d'outils dont l'objectif n'est pas de donner une solution, mais d'aider le décideur dans sa tentative de saisie des préférences entre actions ou entre critères, et dans sa tentative de compréhension et de justification d'un processus qui le conduit finalement à une solution. Souvent, les outils proposés ne s'attachent à résoudre qu'un aspect particulier des difficultés rencontrées.

Le but de ce travail est dès lors de réaliser un logiciel intégrant des outils qui permettront d'assister le décideur dans le processus de saisie de l'information, ainsi que dans le processus de décision. Ce programme doit donc gérer la saisie des données, effectuer les calculs proprement dits, et permettre la sortie de résultats.

Le premier chapitre énumère les contraintes supportées par le logiciel, et le justifie tel qu'il a été conçu.

Parmi les contraintes relevées, les outils qui permettent une saisie plus facile des préférences par l'utilisateur sont développés au Chapitre II.

Le Chapitre III est consacré à la description de méthodes multicritères.

Le Chapitre IV développe la structure et les possibilités du programme (exécution, impression, documentation).

Le Chapitre V est consacré à l'utilisation de certaines méthodes multicritères dans le cadre d'une application : l'aide à l'identification des entérobactéries.

Outre la conclusion, l'Annexe I introduit les notions de base nécessaires à la compréhension du travail relatives aux relations de préférence ainsi qu'aux fonctions d'utilité. L'Annexe II présente les méthodes multicritères non développées au Chapitre II, et qui sont supportées par le logiciel. L'Annexe III fournit une description des fichiers utilisés.

CHAPITRE I.

Chapitre I. Les contraintes imposées au logiciel d'aide à la décision multicritère .

Nous décrivons les contraintes auxquelles le logiciel établi fait face.

Ces limites sont de deux ordres :

- celles associées au caractère multicritère du problème de décision,
- celles liées à la nature des méthodes et à leur utilisation par l'homme.

I.1 Les contraintes associées au caractère multicritère du processus de décision.

I.1.1 La nature du problème de décision.

Soient donnés les ensembles suivants :

- un ensemble K de k_1 critères,
- un ensemble M de m_1 actions.

Cet ensemble M pouvant être a priori de deux types :

- fini : chaque action candidate est identifiée de façon précise,
- infini : l'ensemble des actions est alors celui des solutions d'un système de conditions mises sous forme mathématique et portant sur les caractéristiques des actions candidates,

nous nous sommes intéressés uniquement aux ensembles d'actions finis.

I.1.2 La nature du processus de décision.

Nous appellerons par la suite :

- rapports de préférence partielle entre les actions, l'expression (sous une forme sans importance ici), pour un critère donné, de l'importance relative de chaque action vis-à-vis de ce critère,
- rapports de préférence globale entre les actions, l'expression (sous une forme sans importance ici) de l'importance relative de chaque action vis-à-vis de tous les critères simultanément.

La nature du processus de décision peut être dissociée en deux démarches suivant les objectifs du décideur.

La première attitude du décideur consiste, à partir

- des rapports de préférence partielle entre les actions pour chaque critère
- des rapports de préférence entre les k_1 critères

, à vouloir déterminer les rapports de préférence globale entre les actions; c'est l'approche d'agrégation des rapports de préférence partielle entre les actions.

La seconde attitude du décideur consiste à vouloir réaliser l'approche inverse : à partir des rapports de préférence globale entre les actions, il désire :

- d'une part, retrouver pour chaque critère, les rapports de préférence partielle entre les actions

- d'autre part, obtenir une appréciation de l'importance relative que chaque critère a pris dans la détermination des rapports de préférence globale entre les actions;

c'est l'approche explicative ou de désagrégation des rapports de préférence globale [1], [2], [3], [4].

En fait, ces deux attitudes peuvent être fortement imbriquées.

- Par exemple partant - des rapports de préférence partielle entre actions
 - des rapports de préférence entre critères

le décideur désirera :

1. réaliser une agrégation aboutissant à des rapports de préférence globale
2. les préférences partielles n'étant pas toujours bien fixées dans son esprit, réaliser une désagrégation des préférences globales; en comparant les préférences partielles initiales et celles issues de la désagrégation, il va ajuster ses préférences partielles, puis recommencer le point 1; le processus s'arrêtant lorsque le décideur est satisfait des préférences partielles initiales.

Par conséquent, le logiciel devrait permettre au décideur d'aborder aisément les deux démarches au travers de méthode(s) réalisant les deux approches.

I.1.3 La nature de l'objectif à atteindre dans le processus d'agrégation.

En général, il n'existe pas de solution qui soit optimale simultanément pour tous les critères, certains d'entre eux pouvant être contradictoires; la solution trouvée constituera un compromis. Hormis certains principes généraux tels que :

- le fait que les critères ayant une grande importance aux yeux du décideur seront prépondérants dans le processus d'agrégation
- si une action a de bons rapports de préférence partielle, elle aura un bon rapport de préférence globale
- le respect éventuel de certaines règles comme :
 - . le principe de Pareto : si pour tous les critères, une action a est préférée à une action b, alors l'action a sera globalement préférée à b
 - . le principe de non-dictature : il n'existe pas de critère qui impose systématiquement son classement,

il n'y a pas de règles précises déterminant la façon dont l'agrégation se réalise; par conséquent, il existe de nombreuses méthodes d'agrégation qui se distinguent, en outre, par la manière dont l'agrégation des préférences partielles est obtenue.

Bien souvent, le décideur ne se contente pas de rechercher une seule action comme solution, mais il peut vouloir :

- sélectionner les "bonnes actions"
- obtenir un classement global des actions.

En conséquence, le logiciel comportera plusieurs méthodes d'agrégation qui puissent donner les différents types de résultats.

I.1.4 Les types de processus d'agrégation.

On peut distinguer plusieurs types de traitement de problèmes multicritères en vue de la détermination d'un compromis final parmi un nombre fini d'actions.

I.1.4.1 La construction d'une fonction d'utilité.

Le principe de l'utilisation d'une fonction d'utilité (définie en annexe I) est le suivant [5] :

1. trouver une fonction d'utilité V qui agrège les rapports de préférence partielle entre les k_1 critères,
2. la recherche de l'action optimum revient à chercher, parmi les m_1 actions, celle pour laquelle V est maximum.

I.1.4.2 L'établissement d'une relation de surclassement.

L'objectif est d'obtenir des rapports de préférence globale d'une façon plus souple; la construction d'une fonction d'utilité aboutit, en effet, à définir un préordre global total entre les actions, ce qui suppose et implique :

- que toutes les actions sont comparables, alors que pour certains couples d'actions, l'information disponible au niveau des préférences partielles n'est pas suffisante pour imposer une indifférence ou une préférence globale, et qu'il est souhaitable, alors, de les déclarer incomparables ;
- la transivité des préférences et des indifférences globales.

Principe : considérer chaque couple d'actions séparément; en fonction de la comparaison de leurs rapports de préférence partielle, déterminer le rapport de préférence globale (qui inclut l'incomparabilité).

I.1.4.3 Optimisation en analyse ordinale des données.

A ces types de traitement, il est nécessaire d'en ajouter d'autres, comme ceux associés à l'analyse ordinale des données [29] .

Suivant la nature des données, ce type de traitement concerne :

- les problèmes d'agrégation d'échelles de notes, de valeurs ou de rangs;
- les problèmes de recherche de consensus à distance minimale des préordres initiaux;
- les problèmes de recherche de hiérarchie ou d'ordre total induit par des comparaisons par paires.

Le Logiciel met principalement à la disposition de l'utilisateur des méthodes d'agrégation qui établissent une relation de surclassement; certaines méthodes relatives à l'optimisation en analyse ordinale des données sont disponibles; les méthodes de construction d'une fonction d'utilité sont abordées par le logiciel par la possibilité d'utiliser la méthode "MAL" de Dujmović.

I.2 Les contraintes associées à la nature des méthodes et à leur utilisation par l'homme.

I.2.1 La nécessité d'établir une interface entre l'utilisateur et les méthodes.

Nous avons vu que les méthodes multicritères peuvent utiliser trois types de rapports de préférence :

- les méthodes d'agrégation supposent que le décideur dispose :
 - des rapports de préférence entre les critères,
 - pour chaque critère, des rapports de préférence partielle entre les actions;
- les méthodes de désagrégation nécessitent, pour leur utilisation, la disponibilité :
 - soit uniquement des rapports de préférence globales entre actions,
 - soit :
 - des rapports de préférence globale entre actions,
 - pour chaque critère, des rapports de préférence partielle entre les actions.

De plus, ces méthodes imposent des rapports de préférence exprimés :

- soit sous forme d'un préordre complet (méthodes Electre, Oreste, Qualiflex, UTA, ...)
- soit d'une façon plus contraignante encore, sous forme de nombres appelés poids des critères,

évaluations des actions ou indices d'importance

(méthodes Dujmovic, Saaty, ...).

Or, souvent, l'utilisateur éprouve des difficultés :

- à exprimer les rapports de préférence demandés,
- à les donner sous la forme exigée par la méthode,

et il semble, dès lors, opportun de mettre à la disposition de l'utilisateur des outils qui jouent le rôle d'interface entre lui et la méthode qu'il veut utiliser.

Ainsi, les outils suivants ont été associés au logiciel, et seront expliqués dans le chapitre suivant :

1. pour l'expression des 3 types de rapports de préférence :
 - la possibilité de construire une relation quelconque entre les critères ou les actions, et de l'approximer par un préordre total,
 - la méthode de comparaison hiérarchique par paires de Saaty.
2. pour l'expression des rapports de préférence entre critères.
 - l'utilisation de méthode(s) multicritère(s) explicative(s) ;
3. pour l'expression des rapports de préférence entre les actions pour un critère :
 - la construction de fonctions d'utilité.

I.2.2 La nécessité de pouvoir réaliser des analyses de sensibilité.

Au vu des remarques faites sur les préférences (cfr. point précédent), et sur la nécessité de disposer de plusieurs méthodes d'agrégation (point I.1.3), au vu de l'existence dans les méthodes d'agrégation de paramètres dont on peut faire varier les valeurs, l'utilisateur ne se contentera généralement pas d'une seule exécution d'une seule méthode, mais réalisera plusieurs exécutions de différentes méthodes pour étudier la sensibilité des résultats finaux aux modifications apportées aux rapports de préférence, ou au(x) paramètre(s), ainsi qu'aux différentes méthodes.

Pour faciliter la réalisation des analyses de sensibilité et la réflexion sur les résultats de celles-ci, le logiciel devra assurer l'enregistrement et l'impression des données rentrées par l'utilisateur, ainsi que des résultats obtenus; de plus, la suppression d'enregistrements relatifs à des analyses de sensibilité qui n'ont plus d'intérêt devrait être possible. Ces facilités seront développées dans le chapitre IV.

I.2.3 La nécessité de fournir à l'utilisateur des résultats clairs.

En général, les méthodes d'agrégation sont constituées de deux phases :

- une phase d'agrégation : elle aboutit à une relation de préférence globale,
- une phase de sélection ou de classement des actions, ce qui permet, en outre, de rendre les

résultats plus lisibles que ceux fournis par la relation de préférence.

Si cette seconde phase n'est pas présente dans certaines méthodes, il est nécessaire de l'ajouter.

CHAPITRE II.

Chapitre II Des outils permettant une interface souple entre l'utilisateur et les méthodes multicritères.

Les outils décrits devraient permettre à l'utilisateur :

- dans un premier temps, d'appréhender progressivement ses préférences;
- dans une seconde étape, de les exprimer sous une forme acceptable par la méthode qu'il veut utiliser.

Pour chacun des outils proposés ici, sont décrits le principe, le schéma d'utilisation, les méthodes associées à certaines de ses étapes, ainsi que les éventuelles limites.

II.1 Les outils pouvant être utilisés pour l'établissement des trois rapports de préférence.

II.1.1 La construction d'une relation de préférence quelconque.

II.1.1.1 Principe.

a) Permettre à l'utilisateur d'établir une relation binaire (telle que définie en annexe I) en lui laissant la possibilité d'exprimer entre deux actions ou deux critères :

- soit l'indifférence (I),
- soit la préférence (P),
- soit l'incomparabilité (R).

b) Aboutir, éventuellement par l'utilisation d'une méthode d'approximation de la relation binaire, à une expression des préférences acceptable par l'utilisateur et par la méthode multicritère, c'est-à-dire pour ce qui nous concerne ici, sous forme de préordre total.

II.1.1.2 Le schéma d'utilisation.

1. construction de la relation binaire par l'utilisateur,
2. test du type de la relation construite,
3. si la relation construite est quelconque, approximation de celle-ci par un ou plusieurs préordres totaux,
4. transformation du(des) préordre(s) en rang moyen (tel que défini en annexe I),
5. l'utilisateur choisit l'un des préordres obtenus par approximation ou introduit son propre préordre total.

II.1.1.3 Les méthodes associées à certaines étapes d'utilisation.

II.1.1.3.1 La construction de la relation binaire.

L'utilisation d'un progiciel graphique [9] devrait permettre à l'utilisateur d'avoir une meilleure vue globale des préférences qu'il établit.

II.1.1.3.2 Le test du type de relation et l'approximation d'une relation quelconque.

Des méthodes de test et d'approximation d'une relation binaire sont développées par Jacquet-Lagrèze, par exemple, dans [8]

La définition et les théorèmes suivant justifient ces méthodes.

Définition : soit M ou $\{M_{ij}\}$ la matrice d'adjacence du quasi-ordre Q défini sur un ensemble A de n éléments (cfr. annexe I).
Par définition, le score de l'élément $i \in A$ est :

$$\delta_i = \sum_{j=1}^n (M_{ij} - M_{ji}).$$

Théorèmes :

1. Luce [10] a démontré que, si Q est un quasi-ordre dans A , alors il existe une fonction f définie sur A et à valeurs réelles, et un seuil constant $\epsilon \geq 0$ tels que si $i, j \in A$, on a :

$$\begin{aligned} i P j &\Leftrightarrow f(i) > f(j) + \epsilon, \\ i I j &\Leftrightarrow |f(i) - f(j)| \leq \epsilon. \end{aligned}$$

La relation de préordre (total) T définie par :

$$i T j \Leftrightarrow f(i) \geq f(j)$$

est appelée le préordre sous-jacent à Q . Le seuil ϵ peut s'interpréter comme un seuil d'indifférence (cfr. annexe II.1). Si $\epsilon = 0$, le quasi-ordre Q est un pré-ordre total.

2. Jacquet-Lagrèze [8] a montré que lorsque les lignes et les colonnes de $\{M_{ij}\}$, matrice d'adjacence associée à un quasi-ordre, sont triées suivant les scores décroissants :

1. il existe 2 frontières en escalier symétriques par rapport à la diagonale séparant les couples iP_j , iI_j et jP_i ,
2. le préordre associé aux scores \mathcal{S}_i est le préordre sous-jacent au quasi-ordre.

Remarques :

1. On retrouve les classes d'équivalence du préordre sous-jacent en prolongeant les marches de la frontière en escalier jusqu'à la diagonale.
2. Lorsque les marches de la frontière en escalier s'appuient sur la diagonale, le quasi-ordre est un préordre total.
3. Un ordre total est un quasi-ordre qui comprend n marches s'appuyant sur la diagonale.

Exemple : considérons la matrice d'adjacence suivante associée à un quasi-ordre où

$$iP_j \Leftrightarrow M_{ij} = 1 \text{ et } M_{ji} = 0$$

$$iI_j \Leftrightarrow M_{ij} = 1 \text{ et } M_{ji} = 1$$

	1	2	3	4	5	6	7
1	0	1	1	1	1	1	1
2	1	0	1	1	1	1	1
3	0	0	0	1	1	1	1
4	1	1	1	0	1	1	1
5	0	0	1	0	0	1	0
6	0	0	1	0	1	0	0
7	0	0	1	1	1	1	0

Les scores valent :

$$\mathcal{S}_1 = \mathcal{S}_2 = 4$$

$$\mathcal{S}_3 = -2$$

$$\mathcal{S}_4 = 2$$

$$\mathcal{S}_5 = \mathcal{S}_6 = -4$$

$$\mathcal{S}_7 = 0$$

En ne conservant que la partie asymétrique, et en triant

lignes et colonnes suivant
les δ_i décroissants, on obtient :

	2	1	4	7	3	5	6
2	0	0	0	1	1	1	1
1			0	1	1	1	1
4				0	1	1	
7	0	0			1	1	
3	0	0					
5	0	0	0	0			
6	0	0	0	0			

----- : frontière du quasi-ordre

..... : frontière du préordre
total

II.1.1.3.2.1 L'algorithme de test.

1. Si des incomparabilités ($R \neq \emptyset$) existent dans la relation construite, elle n'est sûrement pas un quasi-ordre.
2. Sinon :
 - calculer le score δ_i pour chaque critère ou action i ,
 - considérer uniquement la partie (P) asymétrique de la relation initiale, et la trier suivant les δ_i décroissants,
 - on vérifie si la frontière séparant les couples (I) des couples (P) est en escalier, c'est-à-dire :
 - \forall indice-ligne i , si iP_j et $i \leq j$
 alors $\forall j' > j$, on doit avoir $iP_{j'}$
 où j et j' sont des indices-colonnes.
 - \forall indice-colonne j , si iP_j et $i \leq j$
 alors $\forall i' < i$, on doit avoir $i'P_j$
 où i et i' sont des indices-lignes.

- si c'est le cas et
 - si pour chaque indice colonne j , l'indice ligne i le plus élevé tel que $M_{ij} = 1$ est égal à $j-1$,
alors la relation est un préordre total
 - sinon la relation est un quasi-ordre
- sinon, la relation est quelconque.

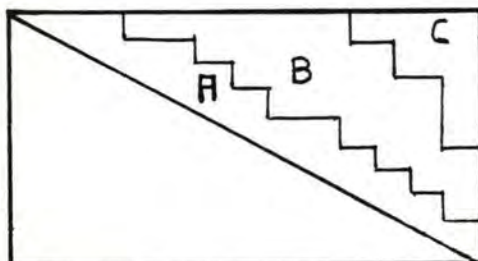
II.1.1.3.2.2 La méthode d'approximation d'une relation quelconque par un quasi-ordre ou d'un quasi-ordre par un préordre total.

1. Si la relation est un quasi-ordre, d'après le théorème 2, le préordre, approximation du quasi-ordre, sera celui associé aux scores δ_i .
2. Sinon : Jacquet-Lagrèze propose une méthode conduisant à trois quasi-ordres; les préordres sous-jacents sont chaque fois déterminés.

Première étape : détermination de 2 quasi-ordres suivant l'algorithme suivant :

1. considérer uniquement la partie (P) asymétrique de la relation initiale.
2. Calculer les scores δ_i , puis trier les lignes et colonnes de la matrice $\{P_{ij}\}$ de P suivant les δ_i décroissants.
3. S'il existe des circuits dans (P) , il existe obligatoirement des "1" sous la diagonale. Supprimer ces 1 (pour éliminer les circuits de P).
4. (P) doit être rendue transitive pour arriver à un quasi-ordre. Sa fermeture transitive est donc établie.
5. Calculer les nouveaux scores δ_i , puis trier P suivant les δ_i décroissants.

Après l'étape 5, on obtient la matrice d'adjacence suivante; elle est divisée



en 3 zones : - la zone C contient uniquement des 1,
 - la zone A contient uniquement des 0,
 - la zone B ou zone d'indétermination contient des 0 ou des 1; elle est comprise entre deux frontières en escalier.

Le premier quasi-ordre, ou quasi-ordre discriminant est obtenu en prenant la frontière inférieure et en remplissant la zone B de 1.

Le second quasi-ordre, ou quasi-ordre non-discriminant, est obtenu en prenant la frontière supérieure et en remplissant la zone B de 0.

Deuxième étape : elle permet de déterminer, suivant une heuristique, un troisième quasi-ordre obtenu en cherchant une frontière intermédiaire à l'intérieur de la zone d'indétermination.

Principe : 2 types de situations empêchent l'existence d'un quasi-ordre.

1ère situation : il existe des $P_{ij} \in B$, égaux à 0 qui couvrent des éléments $P_{i'j'} \in B$, égaux à 1 (figure 1); pour obtenir un quasi-ordre, la solution

serait de mettre P_{ij} à 1.

2ème situation : il existe des $P_{ij} \in B$ égaux à 1 couverts par des éléments $P_{i'j'} \in B$, égaux à 0 (figure 2); pour obtenir un quasi-ordre, P_{ij} devrait être mis à 0.

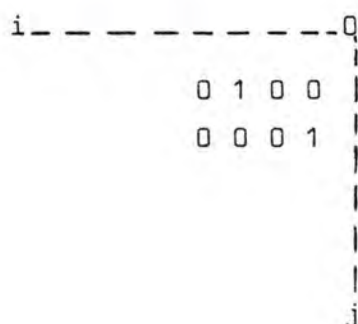


fig. 1

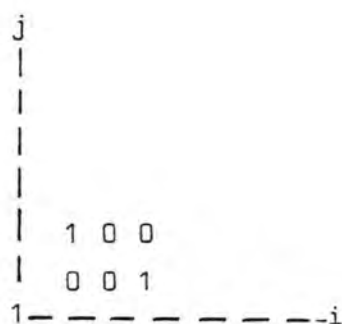


fig. 2

Les 2 situations se présentant simultanément pour plusieurs P_{ij} , l'idée est alors de choisir l'élément P_{ij} qui couvre, ou qui est couvert par le plus grand nombre de $P_{i'j'}$; on change alors la valeur du P_{ij} sélectionné.

L'algorithme est alors le suivant :

6 - $\forall i, j \in B$, on calcule le nombre $So(i, j)$ de 1 mal placés : $So(i, j) = \sum_{i' \geq i} \sum_{j' \leq j} P_{i'j'}$; $j', i' \in B$.

- $\forall i, j \in B$, on calcule le nombre $S1(i, j)$ de 0 mal placés : $S1(i, j) = \sum_{i' \leq i} \sum_{j' \geq j} (1 - P_{i'j'})$; $i', j' \in B$

- soit alors (i_0, j_0) le couple tel que $So(i, j)$ soit maximum,

- soit (i_1, j_1) le couple tel que $S1(i, j)$ soit maximum.

- Si $So(i_0, j_0) = S1(i_1, j_1) = 0$; le 3ème quasi-ordre est obtenu.
- Si $So(i_0, j_0) \geq S1(i_1, j_1)$, $P_{i_0, j_0} = 1$.
- Si $So(i_0, j_0) < S1(i_1, j_1)$, $P_{i_1, j_1} = 0$.

7 On calcule les nouveaux scores S_i , on trie P par ordre des S_i décroissants, on recommence 6 .

Remarque : pour éclairer le choix de l'utilisateur, deux mesures de la qualité de l'approximation de la relation initiale par le quasi-ordre ou/et de la relation initiale par le préordre sous-jacent sont proposées:

1. le coefficient d'approximation : c'est le rapport du nombre de préférences et d'indifférences de la relation initiale conservées dans l'approximation sur le nombre total de cas possibles dans une relation ($= n(n-1)/2$).
2. la distance de Kendall. (cfr. annexe I).

II.1.1.4 Les limites de la construction d'une relation binaire.

La méthode d'approximation d'une relation binaire est d'autant meilleure - en terme des mesures proposées dans la remarque du point précédent - qu'il y a moins de couples incomparables dans la relation construite.

II.1.2 L'utilisation de la méthode hiérarchique des comparaisons par paires de Saaty [14] .

II.1.2.1 Rappel concernant la théorie des matrices.

Les définitions et théorèmes suivants permettent de comprendre le principe de la méthode.

Définition 1 :

la matrice $P(m \times m)$ est positive (ce qui est noté $P > 0$) si $p_{ij} > 0 ; i, j = 1, \dots, m$.

Définition 2 :

soient $\lambda_1, \dots, \lambda_m$ les valeurs propres d'une matrice P ; une valeur propre dominante de P est une valeur propre λ_i de P telle que $|\lambda_i| \geq |\lambda_j| \forall i \neq j$.

Théorème 1 : toute matrice $P > 0$ possède une valeur propre λ_{\max} dominante, réelle, positive ($\lambda_{\max} > 0$), de multiplicité 1 et dont le module est strictement supérieur aux modules des autres valeurs propres de P . De plus, le vecteur propre W correspondant à λ_{\max} est à composantes positives $w_i > 0$ ($i = 1, \dots, m$).

Définition 3 :

une matrice P est réciproque lorsque $p_{ji} = 1/p_{ij} ; i, j = 1, \dots, m$.

conséquence : les éléments diagonaux (p_{ii}) égalent 1.

Théorème 2 : si la matrice $P > 0$ est réciproque et si λ_{\max} est sa valeur propre dominante, alors $\lambda_{\max} \geq m$.

Définition 4 :

une matrice P telle que $p_{ij} \cdot p_{jk} = p_{ik} ; i, j, k = 1, \dots, m$, est appelée matrice consistante.

Théorème 3 : pour qu'une matrice $P > 0$ et réciproque soit consistante, il faut et il suffit que sa valeur propre dominante λ_{\max} soit telle que $\lambda = m$.

Remarques :

- a) les éléments diagonaux d'une matrice consistante sont égaux à 1 puisque pour $j = i$, on a $p_{ii}.p_{ik} = p_{ik} \Rightarrow p_{ii} = 1$
- b) une matrice consistante est toujours réciproque puisque pour $k = i$, on a $p_{ij}.p_{ji} = p_{ii} = 1$
- c) une matrice positive et consistante est toujours positive et réciproque
- d) une matrice positive et réciproque n'est pas nécessairement consistante.

Théorème 4 : soient P une matrice positive et réciproque, λ_{\max} sa valeur propre dominante, et W le vecteur propre correspondant. Alors, $\forall i, j \in \{1, \dots, m\}$
 $(p_{ik} \geq p_{jk}, k = 1, \dots, m) \Rightarrow w_i \geq w_j$.

Théorème 5 : une matrice $m \times m$ de rang $m - s$ ($1 \leq s \leq m$) possède une valeur propre nulle de multiplicité $r \geq s$.

II.1.2.2 Objectif.

Il s'agit de ranger des critères ou des actions au moyen de poids relatifs mesurés sur une échelle de rapport. Ce rangement se fait par l'intermédiaire de comparaisons par paires.

II.1.2.3 Principe.

Première étape : supposons, provisoirement, que nous

disposons des éléments w_i (poids ou évaluations) que nous cherchons, $i = 1, \dots, m$.

On peut se rendre compte de l'importance relative d'un élément i vis-à-vis de tous les autres éléments j en définissant les rapports $\frac{w_i}{w_j}$; si on définit ces rapports pour tous les éléments i , on obtient une matrice P (tableau 1) positive, réciproque et consistante.

D'après le théorème 3, elle possède une valeur propre dominante $\lambda_{\max} = m$.

$P =$

	e1	e2	...	em
e1	w_1/w_1	w_1/w_2	...	w_1/w_m
e2	w_2/w_1	w_2/w_2	...	w_2/w_m
e3	w_3/w_1	w_3/w_2
.
.
em	w_m/w_1	w_m/w_2	...	w_m/w_m

- tableau 1 -

Remarques :

1. si $W = (w_1, \dots, w_m)$, on a $P \cdot W = mW$;

W , que l'on suppose momentanément connu, est donc le vecteur propre de P correspondant à la valeur propre dominante m de P

1. toute colonne i de P est un multiple de W . Dès lors :
 - d'une part, comme les vecteurs propres sont définis à une constante multiplicative près, chaque colonne de P est aussi vecteur propre; pour obtenir l'unicité du vecteur propre, on le normalise de façon à ce que la somme de ses composantes soit égale à 1;

- d'autre part, le rang de P est 1, et le théorème 5 permet d'affirmer que m est la seule valeur propre non nulle.

3. la matrice P recouvre deux formes de consistance :

a) la consistance cardinale : $\forall i, j, k \in \{1, \dots, m\}$:

$$p_{ij} \cdot p_{jk} = p_{ik}$$

b) la consistance ordinale, qui signifie que :

si P_i désigne la ligne i de P , et si

$$P_i \succcurlyeq P_j \iff (\forall k \in \{1, \dots, m\} : p_{ik} \geq p_{jk}),$$

$$P_i \succ P_j \iff P_i \succcurlyeq P_j \text{ et non } (P_j \succcurlyeq P_i),$$

$$P_i \sim P_j \iff P_i \succcurlyeq P_j \text{ et } P_j \succcurlyeq P_i.$$

alors :

$$P_i \succcurlyeq P_j \implies w_i \geq w_j$$

$$P_i \succ P_j \implies w_i > w_j$$

$$P_i \sim P_j \implies w_i = w_j.$$

Seconde étape. Supposons que nous ne disposons pas des éléments w_i , mais que l'utilisateur ait fourni des estimations p_{ij} des rapports w_i/w_j , avec le fait que $p_{ij} = 1/p_{ji}$; en général, la consistance cardinale ne sera plus vérifiée pour la matrice.

Néanmoins :

1. la matrice \tilde{P} des estimations est positive et réciproque, donc, d'après les théorèmes 1 et 2, possède une valeur propre $\tilde{\lambda}_{\max}$ dominante $\geq m$
2. le théorème 4 assure que la consistance ordinale reste vérifiée.

En fonction de la remarque 1, il semble normal, lorsque \tilde{P} n'est pas trop inconsistante - c'est-à-dire lorsque $\tilde{\lambda}_{\max}$ est proche de m -, de choisir comme évaluations des actions ou comme poids des critères, les composantes du vecteur propre normalisé W correspondant à la valeur propre dominante $\tilde{\lambda}_{\max}$.

Remarque : Saaty a établi une mesure de l'inconsistance de P en calculant la valeur $\mu = \frac{\tilde{\lambda}_{\max} - m}{m - 1}$, et en proposant d'accepter la consistance de P tant que $(\mu / 2)^{1/2} \leq 1$.

II.1.2.4 Schéma d'utilisation.

1. Choix d'une échelle de rapports permettant d'exprimer les \tilde{p}_{ij} .
2. Construction de la matrice \tilde{P} par l'utilisateur en fournissant les rapports \tilde{p}_{ij} (pour $i < j$) et en imposant que $\tilde{p}_{ji} = 1/\tilde{p}_{ij}$.
3. Recherche de la valeur propre dominante de \tilde{P} et du vecteur propre correspondant.
4. Comme évaluations ou comme poids, l'utilisateur choisit d'utiliser le vecteur propre, ou d'introduire ses propres valeurs.

II.1.2.5 Explicitation du choix d'une échelle de rapport et de la construction de P.

Pour aider l'utilisateur à définir les rapports \tilde{p}_{ij} , Saaty propose de les exprimer à partir d'une échelle de rapport;

elle traduit le nombre de degré de préférence que l'utilisateur est capable d'exprimer entre toutes paires d'actions ou de critères; les échelles proposées sont les suivantes : 3, 5, 7, 9.

Ainsi, si l'utilisateur choisit l'échelle 5, avec la convention que $\tilde{p}_{ii} = 1$ et $\tilde{p}_{ij} = 1/\tilde{p}_{ji}$; les rapports de préférence \tilde{p}_{ij} qu'il pourra choisir sont les suivants :

Valeur de \tilde{p}_{ij}	Définition	Signification.
1	Dominance égale	w_i et w_j contribuent également à la réa- lisation de l'objec- tif.
3	Dominance moyenne	L'expérience et le jugement fa- vorisent moyennement w_i par rapport à w_j .
5	Dominance forte	w_i est fortement favorisé par rapport à w_j car la pratique montre que w_i domine nettement w_j .
2, 4	Valeurs in- termédiaires entre 2 domi- nances adja- centes.	compromis entre 2 jugements successifs.

II.1.2.6 Les limites de la méthode.

Des expériences psychologiques semblant montrer qu'un individu ne peut comparer simultanément plus de 7 objets (dans une fourchette de ± 2), il est nécessaire, pour avoir une matrice \tilde{P} la moins inconsistante possible, de n'utiliser la méthode que lorsque le nombre de critères ou d'actions est inférieur à 10.

II.2 L'utilisation de méthodes multicritères explicatives pour déterminer les rapports de préférence entre critères.

Certaines méthodes de désagrégation des préférences, comme UTA II, permettent, entre autres, de déterminer l'importance relative des différents critères à partir :

- des rapports de préférence globale entre les actions,
- des rapports de préférence partielle entre les actions pour chaque critère.

La méthode UTA II est développée au chapitre suivant.

II.3 L'utilisation de fonctions d'utilité pour déterminer les évaluations d'actions pour un critère.

II.3.1 Principe.

L'utilisation d'une fonction d'utilité comporte deux étapes :

1. la construction de cette fonction d'utilité,
2. la détermination de l'évaluation de chaque action à partir de la fonction d'utilité.

Pour construire une fonction d'utilité, nous avons choisi la technique du point de valeur médiane [16] . D'autres méthodes, telles que les méthodes d'étude de variation des taux de substitution, exposée par K.R. Oppenheimer [18], qui cherchent les formes explicites des utilités (par exemple $u_i(g_i) = e^{w_i g_i}$ ou $u_i(g_i) = g_i^{w_i}$, avec w_i une constante > 0), les méthodes des loteries équivalentes de P. Fishburn [17], ..., auraient pu être utilisées.

II.3.2 Condition d'utilisation de la technique du point de valeur médiane.

Soit une fonction $y = f(x)$; pour déterminer la valeur d'un objet a pour cette fonction, il est nécessaire de pouvoir associer à a une abscisse x .

Pour pouvoir construire une fonction d'utilité sur un critère suivant la technique du point de valeur médiane, il doit exister pour ce critère une échelle cardinale continue telle que l'abscisse de chaque action puisse y être repérée.

Exemple : dans le problème multicritère "choix d'une voiture", la vitesse maximale, et le confort sont deux des critères de choix;

- pour le critère "vitesse maximale", on sait repérer chaque voiture sur une échelle, par exemple $]0,250]$,
- pour le critère "confort", il n'existe pas de mesure intrinsèque du confort qui permette de repérer les voitures sur une échelle associée à cette mesure.

II.3.3 Principe de la technique du point de valeur médiane.

Supposons donc qu'il existe une échelle cardinale associée au critère i , tel que $[g_{i*}, g_{i*}^*]$ désigne l'intervalle de variation des abscisses des actions sur cette échelle.

L'utilisateur doit alors déterminer le point g_i^* de cet intervalle tel qu'il estime que l'accroissement d'utilité pour passer de g_{i*} à g_i^* égale l'accroissement d'utilité pour passer de g_i^* à g_{i*}^* ; l'utilité de g_{i*} étant nulle et celle de g_{i*}^* étant égale à un par convention, l'utilité de g_i^* sera $1/2$; g_i^* est appelé point de valeur médiane ou point-milieu de l'intervalle $[g_{i*}, g_{i*}^*]$.

Chaque fois qu'un point de valeur médiane est obtenu, le principe peut à nouveau être appliqué sur chacun des deux intervalles qu'il engendre; de nouveaux points de la fonction d'utilité seront ainsi déterminés.

II.3.4 Schéma d'utilisation.

1. Chercher le point de valeur médiane de $[g_{i*}, g_{i*}^*]$;

soit $g.5$ ce point; l'utilité u_i de ce point est :

$$u_i [g.5] = .5.$$

2. Chercher le point de valeur médiane $g.75$ de $[g.5, g_i^*]$ et poser $u_i [g.75] = .75$.
3. Chercher le point de valeur médiane $g.25$ de $[g_i^*, g.5]$ et poser $u_i [g.25] = .25$.
4. Appliquer le test de cohérence (cfr. point II.3.4) au point $g.5$; si ce test est négatif, retourner au point 1.
5. Pour les intervalles $[g_i^1, g_i^2]$ - où g_i^1 et g_i^2 sont des points de valeur médiane adjacents - pour lesquels il s'en estime capable, l'utilisateur détermine le point de valeur médiane $g.m$; l'utilité de $g.m$ sera $1/2 [u_i (g_i^1) + u_i (g_i^2)]$.

Au cours de l'étape 5 :

- si $g.125$ et $g.375$ ont été déterminés, appliquer le test de cohérence au point $g.25$; si le test est négatif, retourner au point 1.
 - si $g.625$ et $g.875$ ont été déterminés, le test de cohérence est appliqué à $g.75$; si le test est négatif, retourner au point 1.
6. Détermination, par extrapolation linéaire, de l'utilité de chaque action, et donc de son évaluation, à partir de son abscisse, des points de valeur médiane et leur utilité.
 7. Comme évaluations des actions, l'utilisateur choisira les valeurs proposées ou les siennes.

II.3.5 Explicitation du test de cohérence.

Si $[g_{i11}, g_i^{12}]$, $[g_{i21}, g_i^{22}]$ sont 2 intervalles pour lesquels les points de valeur médiane g^*1 , g^*2 ont été

déterminés, tels que $u_i(g_{i11}) + u_i(g_i^{12}) = u_i(g_{i21}) + u_i(g_i^{22})$

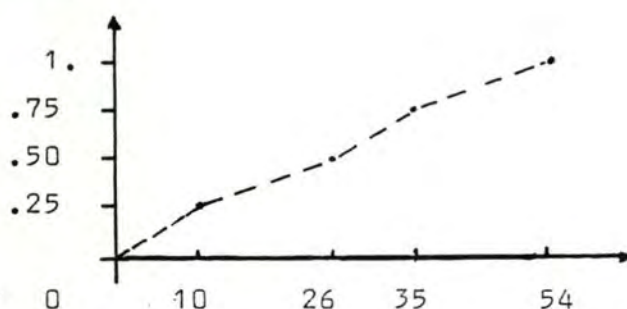
alors,

si l'utilisateur a été précis dans la détermination des points de valeurs médianes g^*1 , g^*2 , on doit avoir

$g^*1 = g^*2$

sinon, on demande à l'utilisateur de réviser son jugement.

exemple :



Si 26 est le point-milieu de $[0, 54]$, alors il doit être aussi le point-milieu de $[10, 35]$.

CHAPITRE III.

Chapitre III. Les méthodes multicritères.

Nous développons des méthodes multicritères associées d'une part au processus d'agrégation et d'autre part au processus de désagrégation.

III.1 Définitions préliminaires.

Plusieurs méthodes utilisent les notions de relation de surclassement déterministe ou de relation de surclassement floue. Avant que ces méthodes ne mettent en évidence les avantages ou les inconvénients de ces relations, donnons-en une idée intuitive.

Une relation de surclassement déterministe S est une relation binaire définie sur M de la manière suivante : étant donné deux actions potentielles a et b appartenant à M,

- "a surclasse b" signifie que, sur base de l'information disponible, on a des raisons suffisantes d'admettre l'hypothèse selon laquelle "a est au moins aussi bonne que b"
- "a ne surclasse pas b" signifie que sur base de l'information disponible, les arguments en faveur de la proposition "a est au moins aussi bonne que b" sont jugés insuffisants.

Par conséquent, le surclassement simple entre deux actions correspond à la situation où une action est préférée à une autre; le surclassement double implique l'indifférence entre les deux actions; l'absence de surclassement implique l'incomparabilité entre les deux actions.

Une relation de surclassement floue S_f est caractérisée par la définition d'un degré de crédibilité d associant à tout couple (a, a') un nombre $d(a, a')$; d étant un critère destiné à repérer la plus ou moins grande crédibilité du surclassement de a' par a , il doit posséder les propriétés suivantes :

propriété 1 : le nombre $d(a', a)$ ne fait intervenir a' et a qu'au travers de leurs évaluations sur l'ensemble des critères; ainsi, si $X(a) = (X_1(a), \dots, X_k(a))$ est un vecteur associé à l'action a où $X_i(a)$ désigne l'évaluation de l'action a pour le critère i , on a $d(a', a) = d[X(a'), X(a)]$.

propriété 2 : $d(a', a)$ est d'autant plus grand que la fiabilité du surclassement de a par a' est plus grande, donc, en particulier : $d(a', a)$ est une fonction non décroissante de $X_i(a')$ $\forall i \in K$ et non croissante de $X_i(a)$ $\forall i \in K$.

propriété 3 : $d(a', a) = 1$ traduit un surclassement certain de a par a' , alors que $d(a', a) = 0$ traduit, soit un non surclassement certain de a par a' , soit l'absence totale de preuves en faveur d'un tel surclassement; il s'ensuit que :

$$0 \leq d(a', a) \leq 1.$$

Remarque : le degré de crédibilité ne doit pas être confondu avec un indicateur d'intensité de préférence.

III.2 La méthode Oreste.

III.2.1 Objectif de la méthode.

La méthode Oreste [11], [13] suppose que :

- pour chaque critère k , les rapports de préférence partielle entre les m_1 actions sont exprimés à l'aide d'un préordre total ; le rang moyen de chaque action a est alors $rk(a)$, $1 \leq rk(a) \leq m_1$;
- les rapports de préférence entre les k critères sont exprimés grâce à un quasi-ordre que l'on peut ajuster à un préordre total ; cet ajustement conduit au rang moyen du critère k : rk , $1 \leq rk \leq k_1$.

A partir de ces préordres, l'objectif est d'obtenir une relation de surclassement déterministe d'après l'idée suivante :

chaque couple (action a , critère k), noté ak , est projeté sur une demi-droite à partir de laquelle on attribue un rang moyen à chaque point ak projeté, pour ensuite déterminer la relation de surclassement.

Les étapes de la procédure sont donc au nombre de trois :

1. la phase de projection sur D ,
2. la phase de rangement,
3. la phase d'agrégation.

III.2.2 Explicitation de ces étapes.

III.2.2.1 La phase de projection sur D.

La projection de a_k sur D , demi-droite d'origine \emptyset , dépend de 2 éléments : les rangs $rk(a)$ et rk . Une formule devra donc, à partir de ces 2 éléments, déterminer la position de a_k en terme de distance par rapport à \emptyset ; les formules proposées sont des métriques d qui doivent être telles que

$$d(\emptyset, a_k) < d(\emptyset, b_k) \text{ si } a P_k b \text{ pour le critère } k.$$

Deux distances adéquates sont suggérées par Roubens :

1. une distance du type city-block :

$$d(\emptyset, a_k) = \alpha . rk(a) + (1 - \alpha) rk, \quad 1 \leq \alpha \leq 1;$$

α est un paramètre qui traduit dans quelle mesure les rangs $rk(a)$ et rk interviennent, chacun, dans la détermination de la position qu'occupera a_k sur la demi-droite D .

Remarque : Lorsque k_1 est différent de m_1 , les intervalles de variation des rangs rk et $rk(a)$ sont différents; la distance city-block modifiée d' peut alors être considérée :

$$d'(\emptyset, a_k) = \alpha k_1 rk(a) + (1 - \alpha) m_1 rk.$$

2. La distance de Hölder :

$$d_R(\emptyset, a_k) = \left\{ \frac{1}{2} [r_k(a)]^R + \frac{1}{2} [r_k]^R \right\}^{1/R}$$

où R est un paramètre à fixer.

III.2.2.2 La phase de rangement.

Un rang moyen est attribué à chaque point a_k tel que
 $R(a_k) \leq R(b_j)$ ssi $d(\emptyset, a_k) \leq d(\emptyset, b_j)$
 avec $1 \leq R(a_k) \leq m_1 k_1$.

III.2.2.3 La phase d'agrégation.

Ayant attribué un rang $r(a_k)$ à chaque a_k , on peut définir le rang global de chaque action : $R(a) = \sum_{k=1}^{k_1} R(a_k)$;

on obtient ainsi un préordre global entre les actions; cependant, la relation de surclassement ne peut être réduite à ce préordre car celui-ci ne tient pas compte des situations suivantes :

soient a et b 2 actions; $R(a)$ et $R(b)$ leurs rangs globaux respectifs.

Supposons $R(a) \leq R(b)$.

Première situation pouvant se présenter :

$R(a)$ et $R(b)$ sont "si proches" qu'on peut déclarer a et b indifférents.

Pour mesurer cette proximité, on détermine le rapport

$$\frac{R(b) - R(a)}{\quad}$$

(différence maximale entre les rangs de deux actions.)

Cette différence maximale est atteinte pour 2 actions c et d dans le cas où $R(d)$ est maximum, et $R(c)$ est minimum; c'est-à-dire quand :

$$- \forall k, r(d_k) \text{ est maximum, c'est-à-dire } r(d_k) = m_1 k_1 - \frac{k_1 + 1}{2}$$

- $\forall k, r(ck)$ est minimum, c'est-à-dire $r(ck) = \frac{k1 - 1}{2}$.

A ce moment, $R(d) - R(c) = k1(m1-1)$.

On doit de plus introduire un seuil, appelé seuil d'indifférence, $\beta \in [0,1]$ tel que :

- si $\frac{R(b) - R(a)}{k1(m1-1)} \leq \beta$ alors aIb

- sinon a n'est pas jugé suffisamment proche de b que pour les déclarer indifférents; à ce moment,

la seconde situation peut se présenter :

si on définit $C(a,b) = \sum_{k: aP_k b} [R(bk) - R(ak)] \quad (> 0)$

($C(b,a)$ est défini de façon analogue),

$\forall k$, $R(bk)$ et $R(ak)$ peuvent être tels que $C(b,a)$ soit "si élevé" qu'il est dangereux de déclarer aPb .

Pour mesurer ce danger, on calcule le rapport $\frac{C(b,a)}{R(b) - R(a)}$;

si ce rapport est "suffisamment élevé", on ne pourra admettre sans risque que aPb , et on déclarera a et b incomparables.

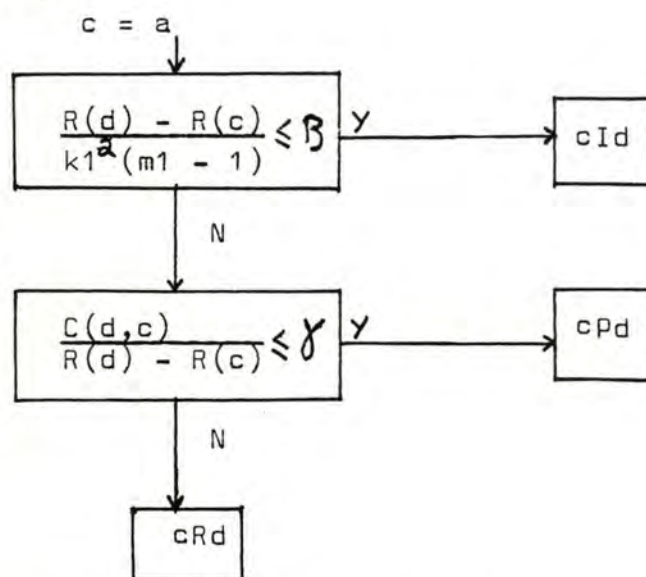
On doit par conséquent, introduire un seuil d'incomparabilité $\gamma \geq 0$ tel que :

- si $\frac{C(b,a)}{R(b) - R(a)}$ est supérieur à γ , alors aRb

- sinon aPb .

La procédure d'agrégation est alors immédiate : pour chaque couple (a,b) d'actions,

- on calcule $R(a)$, $R(b)$, $C(b,a)$
- si $R(a) \leq R(b)$, on applique l'algorithme suivant,
pour $d = b$



- si $R(a) > R(b)$, on applique l'algorithme précédent
pour $d = a$
 $c = b$.

Remarques :

1. Pour respecter le principe de Pareto, β doit être choisi $< 1/k_1 (m_1 - 1)$.
2. Lorsque l'utilisateur dispose d'évaluations et de poids précis, plutôt que de préordres, il est plus intéressant de pouvoir utiliser directement ces poids et ces évaluations que de passer par la transformation en rang moyen des préordres qui leurs sont associés. Ceci est permis par le logiciel. La seule précaution à prendre, c'est de normaliser (sur 1, par exemple) le vecteur des poids et chacun des k_1 vecteurs des évaluations avant la phase de projection; en effet, aux différents critères, sont souvent associées des échelles de mesure différentes; une non-

normalisation fausserait alors le calcul de la distance dans la phase de projection.

3. Lorsque la relation de surclassement établie est trop dense, il est difficile, pour l'utilisateur, de l'exploiter. Pour cette raison, le logiciel offre deux possibilités:

1. déterminer, si la relation de surclassement n'est pas un quasi-ordre, trois quasi-ordres proches de celle-ci, et ce en appliquant les méthodes de tests et d'approximation de relations (cfr. II.1.1)
2. déterminer les "bonnes actions" de la relation de surclassement en appliquant la méthode de recherche des quasi-noyaux associée à la méthode multi-critère Electre-I, et qui est développée avec celle-ci au point III.4.2.2.3.

4. Les paramètres qui pourront faire l'objet d'une analyse de sensibilité sont :

- le seuil d'indifférence B
- le seuil d'incomparabilité γ
- la distance ou/et le paramètre qui lui est associé.

III.3 Des méthodes explicatives d'une préférence globale : les méthodes UTA.

III.3.1 Objectif de la méthode.

Les méthodes UTA [1], [2], [3], [4], cherchent les fonctions d'utilités additives $U(g) = \sum_{i=1}^{k-1} u_i(g_i)$, où

$u_i(g_i)$ est la fonction d'utilité partielle associée au critère i , qui expliquent "au mieux" la relation de préférence globale R entre les actions exprimée par le décideur.

Remarques :

1. Le pouvoir explicatif de chaque fonction d'utilité $U(g)$ est déterminé ainsi :

de $U(g)$, on déduit un préordre global R^* sur les actions défini comme suit :

$\forall a, b \in M$: si $U[g(a)] = U[g(b)]$, alors a et b appartiennent à la même classe d'indifférence de R^*

si $U[g(a)] > U[g(b)]$, alors a est préféré à b dans R^* .

On calcule, alors, l'indicateur τ de Kendall entre R et R^* ; plus τ sera proche de 1, et meilleure sera la représentation de la relation globale par $U(g)$.

2. Il existe plusieurs méthodes UTA; les 2 méthodes de base, appelées UTA I et UTA II, développées ci-dessous supposent que la relation de préférence globale est un préordre constitué de Q classes d'indifférence; chacune de ces classes i a n_i éléments.

3. Chacune des méthodes procède en 3 étapes :

la première étape consiste à rechercher, par résolution d'un programme linéaire, la "fonction d'utilité optimale" compte tenu des contraintes de forme imposées aux utilités partielles et la contrainte de bonne représentation des préférences globales par cette fonction $U(g)$.

De par la nature de $U(g)$, de la relation de préférence globale et des contraintes supplémentaires imposées par le décideur, la recherche d'une seule fonction d'utilité n'est pas suffisante; d'autres fonctions d'utilités seront recherchées au voisinage de $U(g)$. C'est l'objet de la seconde étape, appelée analyse post-optimale.

La troisième étape permet, à partir des n fonctions d'utilités trouvées, de définir des relations de surclassement basées sur la concordance des fonctions d'utilités.

Les fonctions d'utilités associées à UTA I et UTA II étant distinctes, le principe de l'étape 1 est propre à chaque méthode.

III.3.2 La construction de la fonction d'utilité pour la méthode UTA I.

Le problème résolu par la méthode UTA I est celui de déterminer les modèles linéaires $U(g) = u_1(g_1) + \dots + u_{k_1}(g_{k_1})$, en recherchant la forme des fonctions d'utilité partielle $u_i(g_i)$, $i = 1, \dots, k_1$.

III.3.2.1 Condition d'utilisation de UTA I.

Elle est similaire à celle imposée (cfr. point II.3.1) pour construire une fonction d'utilité pour déterminer

les évaluations d'actions pour un critère : pour chaque critère on doit pouvoir repérer l'abscisse $g_i(a)$ de chaque action a sur une échelle G_i (discrète ou continue); soit donc $[g_{i*}, g_{i*}]$, $i = 1, \dots, k$, les domaines de variation de ces abscisses où g_{i*} est l'abscisse du point le moins préféré et g_{i*} celle du point le plus préféré.

III.3.2.2 La méthode de construction de $u_i(g_i)$.

$u_i(g_i)$ est construit en estimant les utilités d'un nombre α_i de ses points; L'abscisse g_i^j de ces points est alors donnée par la formule

$$g_i^j = g_{i*} + \frac{j-1}{\alpha_i-1} (g_{i*} - g_{i*}).$$

Remarques :

1. Si l'échelle est discrète, α_i sera posé égale au nombre d'échelons.
2. Pour chaque critère i et pour chaque action a , si $g_i(a) \in [g_i^j, g_i^{j+1}]$, alors l'utilité partielle de cette action sera donnée, par extrapolation linéaire, par

$$u_i [g_i(a)] = u_i(g_i^j) + \frac{g_i(a) - g_i^j}{g_i^{j+1} - g_i^j} [u_i(g_i^{j+1}) - u_i(g_i^j)].$$

III.3.2.3 Les hypothèses faites sur les utilités partielles $u_i(g_i)$.

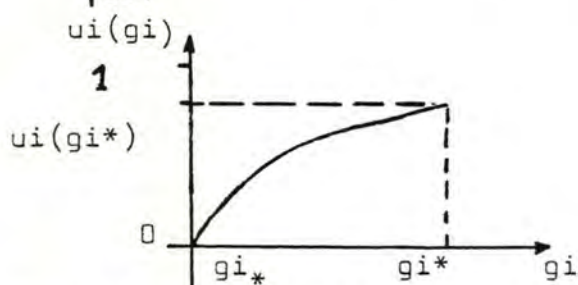
Ces hypothèses sont résumées dans la figure 1; ce sont :

1. $u_i(g_i)$ est une fonction monotone non décroissante des g_i , ce qui se traduira par les contraintes dont l'équation est :

$u_i(g_i^{j+1}) - u_i(g_i^j) \geq s_i \quad \forall i, j \quad (1)$ où $s_i (\geq 0)$, appelé seuil d'indifférence, est à fixer pour chaque critère.

2. Les contraintes de normalisation les plus courantes [16] sont :

$$\begin{cases} u_i(g_{i*}) = 0 \\ \sum_{i=1}^{k1} u_i(g_i^*) = 1 \end{cases} \quad (2) \quad \forall i.$$



- figure 1 -

III.3.2.4 La détermination de la fonction d'utilité $U(q)$.

La fonction d'utilité U à trouver doit expliquer parfaitement R ; ce qui signifie qu'elle doit respecter les indifférences et les préférences entre actions de $R \iff$

$$\forall a, b \in M, U[g(a)] > U[g(b)] \iff a P b \quad (4)$$

$$\text{ou} \quad U[g(a)] = U[g(b)] \iff a I b \quad (5).$$

Compte tenu des contraintes imposées, la fonction d'utilité U' trouvée risque de ne pas expliquer parfaitement R ; à ce moment, la fonction d'utilité U' appliquée à chaque action présentera une erreur relative $\delta(a) > 0$ par rapport à l'utilité U

$$\begin{aligned} U'[g(a)] &= U[g(a)] + \delta(a) \\ &= \sum_{i=1}^{k1} u_i[g_i(a)] + \delta(a), \forall a. \end{aligned}$$

Remarquons que l'on a intérêt à voir $\delta(a)$ tendre vers 0, $\forall a$; donc, minimiser la somme des erreurs relatives peut

constituer la fonction objectif du programme linéaire.

Les contraintes (4) et (5) s'imposant à U , $\forall a, b \in M$,

$$aIb \Leftrightarrow \sum_{i=1}^{k1} (u_i [g_i(a)] - u_i [g_i(b)]) + G(a) - G(b) = 0 \quad (6)$$

$$aPb \Leftrightarrow \sum_{i=1}^{k1} (u_i [g_i(a)] - u_i [g_i(b)]) + G(a) - G(b) \geq \delta \quad (7)$$

où δ est un seuil ≥ 0 traduisant la préférence de a sur b , il s'interprète comme la plus petite différence d'utilité que l'on souhaite restituer entre deux classes consécutives du préordre R .

Si $a, a', a'' \in M$, remarquons que :

1. si la contrainte (6) a été exprimée pour aIa' et pour $a'Ia''$, alors elle est vérifiée pour aIa'' ,
2. de même, si la contrainte (7) a été exprimée pour aPa' et pour $a'Pa''$, alors elle est vérifiée pour aPa'' .

Par conséquent,

1. dans une même classe d'équivalence, la contrainte (6) ne sera exprimée que pour aIa' et $a'Ia''$;
2. la contrainte (7) ne sera exprimée qu'entre classes d'équivalence adjacentes, et qu'une seule fois par classes d'équivalence adjacentes.

Du fait des erreurs $G(a)$, les contraintes (4) et (5) risquent de ne plus être vérifiées par U ; le préordre initial R et le préordre R^* sous-jacent peuvent alors être différents.

Utilisant la remarque 2 du point II.3.2.2, chacune des contraintes (6) et (7) possibles est transformée pour être exprimée en terme d'utilité partielle $u_i(g_i^j)$

$\forall i, j$.

Toutes les contraintes ayant été exprimées, une fonction objectif ayant été énoncée, les utilités $u_i(g_i^j)$ peuvent être déterminées par résolution du programme linéaire suivant :

$$\begin{array}{l}
 \text{PL 1} \left\{ \begin{array}{l}
 [\min] F = \sum_{a \in K} G(a) \\
 \text{sous les contraintes :} \\
 \sum_{i=1}^{k1} (u_i [g_i(a)] - u_i [g_i(b)]) + G(a) - G(b) \geq \delta \quad \text{si } a P b (7) \\
 \sum_{i=1}^{k1} (u_i [g_i(a)] - u_i [g_i(b)]) + G(a) - G(b) = 0 \quad \text{si } a I b (6) \\
 u_i(g_i^{j+1}) - u_i(g_i^j) \geq s_i \quad \forall i, j \quad (1) \\
 \sum_{i=1}^{k1} u_i(g_i^*) = 1 \quad (2) \\
 u_i(g_{i*}) = 0, u_i(g_i^j) \geq 0 \quad \forall i, j \text{ et } \forall a \in M.
 \end{array} \right.
 \end{array}$$

Remarques : 1. les dimensions de PL1 sont les suivantes :

Contraintes			Variables	
Formule	Signe	Nombre	Nature	Nombre
(7)	\geq	$Q - 1$	$u_i(g_i^j)$	$\sum_{i=1}^{k1} (\alpha_{i-1})$
(6)	\geq	$2 \times \sum_{i=1}^Q (n_i - 1)$		
(1)	\geq	$\sum_{i=1}^{k1} (\alpha_{i-1})$	$G(a)$	$m1$
(2)	\geq	2		

Vu ces dimensions, on a intérêt à considérer le dual [19] de PL1 pour déterminer les $u_i(g_i^j)$.

2. Les paramètres pouvant faire l'objet d'une analyse de sensibilité sont :

- le seuil δ
- le nombre de points estimés, α_i , pour chaque fonction d'utilité,
- les seuils d'indifférence s_i .

III.3.3 La construction de la fonction d'utilité pour la méthode UTA II.

Une forme pondérée de la fonction d'utilité additive est donnée par $V(g) = \sum_{i=1}^{k1} p_i u_i(g_i)$ où p_i est le poids

du critère i .

La méthode UTA II détermine alors les fonctions $V(g)$ qui expliquent le mieux la préférence globale en recherchant les vecteurs de poids $(p_1; \dots; p_{k1})$ correspondants.

Elle suppose, dès lors, que les utilités partielles $u^i(g_i)$ sont connues; c'est le cas lorsque l'utilisateur dispose pour chaque critère des évaluations normalisées des actions; celles-ci interprétées comme étant les utilités partielles des actions, représentent une fonction d'utilité implicite.

Les contraintes associées à $V(g)$ sont de même nature que celles développées au point III.3.2.4 pour UTA I, leur expression diffère de par la disponibilité des utilités partielles des actions $u^i [g_i(a)]$.

Les contraintes (4) et (5) deviennent pour UTA II :

$$\sum_{i=1}^{k1} [u^i [g_i(a)] - u^i [g_i(b)]] \cdot p_i + G(a) - G(b) \geq \delta$$

si aPb (8)

$$\sum_{i=1}^{k1} [u^i [g_i(a)] - u^i [g_i(b)]] \cdot p_i + G(a) - G(b) = 0$$

si aIb (9)

Le vecteur de poids (p_1, \dots, p_{k1}) normalisé sur 1 est déterminé par résolution du programme linéaire suivant :

$$\begin{array}{l}
 \text{PL2} \left\{ \begin{array}{l}
 [\min] F = \sum_{a \in K} G(a) \\
 \text{sous les contraintes :} \\
 \sum_{i=1}^{k1} (u'_i [g_i(a)] - u'_i [g_i(b)]) \cdot p_i + G(a) - G(b) \geq \delta \\
 \text{si } aPb \text{ (8)} \\
 \\
 \sum_{i=1}^{k1} (u'_i [g_i(a)] - u'_i [g_i(b)]) \cdot p_i + G(a) - G(b) \\
 = 0 \text{ si } aIb \text{ (9)} \\
 \\
 \sum_{i=1}^{k1} p_i = 1 \\
 \\
 p_i \geq 0, i=1, \dots, k1; G(a) \geq 0 \quad \forall a \in M.
 \end{array} \right.
 \end{array}$$

Remarques :

1. Les dimensions de PL2 sont plus réduites, puisque son dual ne contient que $k1 + m1$ contraintes de signe " \leq " à $(2 \times \sum_{i=1}^Q (n_i - 1) + Q + 1)$ variables.
2. Pour passer de la fonction d'utilité de la première forme U à celle de la forme pondérée, il suffit de poser $u'_i(g_i) = \frac{1}{p_i} (u_i(g_i))$;
en particulier, $u_i(g_i^*) = p_i$.
3. Le seuil δ pourra faire l'objet d'une analyse de sensibilité.

III.3.4 L'analyse post-optimale.

III.3.4.1 Justification de cette étape.

L'existence de cette étape est liée aux problèmes que pose la construction d'une fonction d'utilité comme modélisation des préférences d'un décideur exprimées sous forme de relation; ces problèmes sont les suivants :

1. la fonction d'utilité est peu stable.

Cette instabilité trouve son origine dans :

- la nature de la préférence globale : l'information apportée par la relation globale est très pauvre puisque seulement traduite sous forme de préférence ou d'indifférence entre actions .
- la multiplicité des dimensions exigées des programmes linéaires
- les phénomènes de forte corrélation entre critères qui peuvent fausser et altérer une bonne explication de la préférence globale
- $U(g)$ ou $V(g)$ n'est pas toujours optimale par rapport à des mesures telles que le τ de Kendall entre R et R^* ; d'où intérêt de rechercher au voisinage de $U(g)$ ou de $V(g)$ les fonctions d'utilité qui ont un τ de Kendall au moins aussi bon.

- ##### 2. En plus de la relation de préférence globale, l'utilisateur peut donner des précisions sur l'importance - faible ou forte - qu'il accorde à certains critères. Quand on sait qu'une fonction d'utilité peut être considérablement modifiée en changeant l'importance accordée à un ou plusieurs critères, il est nécessaire de connaître la forme des fonctions

d'utilités associées à cette information supplémentaire.

Pour toutes ces raisons, il est nécessaire de rechercher d'autres fonctions d'utilité se trouvant au voisinage de la solution optimale.

III.3.4.2 Principe de l'analyse post-optimale.

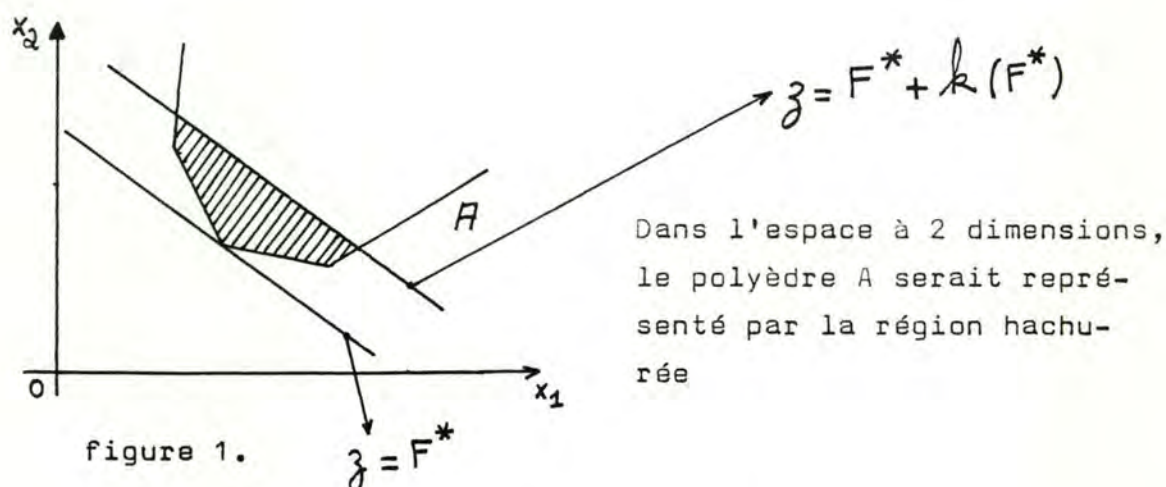
Si on considère :

1. le programme linéaire PL3 suivant :

$$\left\{ \begin{array}{l} \min z = cx \\ \text{sous les contraintes} \\ Ax \geq b \\ x \geq 0 \end{array} \right\} (1)$$

où : A est le tableau du simplex,
 x est le vecteur des inconnues,
 b le second membre du PL,
 c le vecteur des coefficients des x_i ,
 auquel est associée la valeur optimale F^* .

2. Un seuil $k(F^*) \geq 0$, réel "assez petit" dépendant de F^* , l'analyse post-optimale consiste à explorer les sommets du polyèdre A - cfr. figure 1 - obtenu en ajoutant la contrainte $z \leq F^* + k(F^*)$ (2) aux contraintes (1).
 Ces sommets sont des solutions de PL3 légèrement moins optimales que F^* .



III.3.4.3 Application aux méthodes UTA.

La recherche de tous les sommets du polyèdre A étant très coûteuse, les méthodes UTA se contentent d'explorer les sommets aussi différents les uns des autres que possible, et plus précisément ceux correspondant à des fonctions d'utilités qui attribuent des poids extrémaux à un ou plusieurs critères à la fois; cette exploration partielle du polyèdre A est effectuée par la résolution successive des programmes linéaires définis par les contraintes de UTA I ou de UTA II et la contrainte (2), auxquels on associe la fonction objectif :

$$\begin{aligned}
 & [\min] \text{ ou } [\max] \sum_i \rho_i \cdot p_i \text{ pour UTA II,} \\
 \text{ou } & [\min] \text{ ou } [\max] \sum_i \rho_i \cdot u_i(g_i^*) \text{ pour UTA I.}
 \end{aligned}$$

Le schéma d'utilisation est le suivant :

1. définir $k(F^*)$
2. définir $\rho_i, \forall i \in I$ $\rho_i = 1$
 si l'utilisateur veut déterminer la fonction d'utilité minimisant ou maximisant ce critère i (éventuellement simultanément avec d'autres critères)

3. Définir l'objectif : minimiser ou maximiser.
4. Résolution du PL.
5. Etude du pouvoir explicatif de la fonction d'utilité trouvée.
6. Retourner à l'étape 1 ou 2.

Ce procédé permet de donner une idée :

- de la sensibilité des fonctions d'utilités établies aux variations des poids ,
- de la fourchette de variation des poids d'un ou de plusieurs critères à la fois. Si, par exemple, on se donne successivement les fonctions objectives $[\min]P_i$ et $[\max]P_i$, on peut conclure l'importance que le décideur accorde au critère i .

III.3.5 Etablissement d'une relation de surclassement.

L'analyse post-optimale permet à l'utilisateur de déterminer un ensemble de fonctions d'utilité.

Si $U = \{U_1, \dots, U_n\}$ ($n \geq 2$)

constitue un système de fonctions d'utilités retenues par l'utilisateur parce qu'elles sont suffisamment consistantes, en terme de \uparrow de Kendall, avec le préordre initial, pour déterminer les actions les mieux classées sur les n préordres sous-jacents aux fonctions d'utilité, il faut synthétiser ces n préordres en une relation de surclassement qui, confrontée aux préférences initiales du décideur, sera un élément de discussion pour déterminer le classement global final.

Une relation de surclassement floue peut être établie;

elle est basée sur le calcul du pourcentage de fonctions d'utilité U_i pour lesquelles a est meilleure que b ; pour cela, on définit l'indicateur binaire suivant :

$$\forall a, b \in M, \phi_i(a, b) = \begin{cases} 1 & \text{si } U_i(a) \geq U_i(b) \\ 0 & \text{sinon.} \end{cases}$$

Le degré de crédibilité est alors : $d(a, b) = \frac{1}{n} \sum_{i=1}^n \phi_i(a, b)$.

On remarque que $d(a, b) + d(b, a) = (1 + \text{nombre de fois pour lesquels } a \text{ et } b \text{ sont indifférents})/n$.

Cette propriété de $d(a, b)$ nous incite à dire que 2 actions a et b sont incomparables quand $d(a, b)$ et $d(b, a)$ sont inférieurs à un certain seuil $\lambda > .5$ (par exemple $\lambda = .6$); on définit ainsi une relation de surclassement non-floue avec seuil, donnée par la formule suivante :

$$d^\lambda(a, b) = \begin{cases} d(a, b) & \text{si } d(a, b) \geq \lambda \\ 0 & \text{sinon.} \end{cases}$$

La relation ainsi définie est moins dense et plus lisible que la relation de surclassement floue.

III.4 Les méthodes Electre.

Les trois méthodes Electre (Elimination et choix traduisant la réalité) que nous présentons supposent que :

- pour chaque critère k , les rapports de préférence partielle entre les m actions sont exprimés à l'aide d'un préordre total ; à chaque action a est alors associé un nombre $X_k(a)$, appelé évaluation de l'action a pour le critère k , tel que plus $X_k(a)$ est grand, plus la préférence du décideur est grande pour a vis-à-vis du critère k ;
- les rapports de préférence entre les k critères sont exprimés à l'aide d'un poids $p_k > 0$ associé à chaque critère k ; ces poids sont tels que :

$$\sum_k p_k = 1.$$

Les méthodes Electre s'attachent :

1. à la construction de la relation de surclassement à partir des p_k et des $X_k(a)$;
2. à partir de la relation de surclassement, et suivant les objectifs du décideur,
 - à la sélection des "bonnes actions" : c'est la problématique d'Electre I
 - au classement global des actions : ceci est réalisé par Electre II et Electre III d'après l'idée suivante :
 les évaluations $X_k(a)$ des actions étant généralement imprécises et souvent conflictuelles, on peut douter de la possibilité de dégager, dans tous les cas, d'un préordre complet unique réellement significatif. Ces deux méthodes vont donc établir deux préordres complets fondés sur des considérations opposées;

les préordres obtenus seront comparés pour voir s'ils sont suffisamment proches que pour élaborer un classement final.

III.4.1 Le principe de construction des relations de surclassement.

La construction et la nature des relations de surclassement sont distinctes pour chacune des 3 méthodes. Cependant, elles sont établies sur base d'idées communes : les relations de surclassement sont construites en comparant les actions par paires (a,b); on décide que a surclasse b si a est préféré à b "relativement à une majorité de critères sans pour autant être significativement plus mauvais que b."

Les méthodes procèdent donc à un double filtrage pour légitimer le surclassement de a sur b:

1. concordance pour une majorité suffisante de critères.
2. non discordance trop importante au niveau des critères k où $b P_k a$.

Remarque : pour Electre I et Electre II, les 2 filtres impliquent la définition :

1. d'un indice de concordance $C(a,b)$ égal au pourcentage des poids des critères pour lesquels b n'est pas préféré à a; pour mesurer si la concordance est suffisante, on introduit un(des) seuil(s) de concordance $C \in]0,1[$
tel que :
si $C(a,b) < C$,
alors a ne surclasse pas b;

2. d'un seuil de discordance $D_k > 0$ associé à chaque critère k tel que si $X_k(b) - X_k(a) > D_k$, alors la discordance est trop importante pour admettre que a surclasse b .

III.4.2 La méthode Electre I [11], [20].

III.4.2.1 Principe de la construction de la relation de surclassement.

La relation de surclassement déterministe est établie ainsi :

si le décideur a choisi - un seuil de concordance C
- un seuil de discordance D_k

$$\forall k \in K,$$

on considère chaque couple d'actions (a, b) :

étant donné que :

$I^+(a, b)$ est l'ensemble des indices des critères par rapport auxquels a est préféré strictement à b ;

$I^=(a, b)$ est l'ensemble des indices des critères par rapport auxquels a et b sont indifférents;

$I^-(a, b)$ est l'ensemble des indices des critères par rapport auxquels b est strictement préféré à a ;

$$P^+(a, b) = \sum_{i \in I^+(a, b)} p_i, \quad P^-(a, b) = \sum_{i \in I^-(a, b)} p_i$$

$$P^-(a, b) = \sum_{i \in I^-(a, b)} p_i$$

$$C(a, b) = \frac{P^+(a, b) + P^-(a, b)}{P^+(a, b) + P^-(a, b) + P^-(a, b)} = P^+(a, b) + P^-(a, b),$$

si $C(a, b) \geq C$ et $P^+(a, b) \geq P^-(a, b)$

et $\forall i \in I^+ \{ X_i(b) - X_i(a) > 0, X_i(b) - X_i(a) \leq D_i$

alors la relation a surclasse b (aSb) peut être établie.

Remarque : dans certains problèmes, la contribution de $P^=(a,b)$ dans $C(a,b)$ peut être très importante, ce qui peut avoir pour conséquence d'obtenir une relation de surclassement peu discriminante; pour réduire le nombre d'arcs, on peut introduire comme test de concordance $C(a,b) > C$ à la place de $C(a,b) \geq C$.

III.4.2.2 La problématique de la sélection des "bonnes actions".

En général, la relation de surclassement S est construite de telle sorte que pour une action a , tantôt il existe des actions qui lui sont incomparables, tantôt il existe des actions qui la surclassent; par conséquent, il est souvent impossible de trouver la "bonne action a " telle que $\forall b \in M : aSb$.

Roy a alors suggéré de chercher un sous-ensemble d'actions ayant des propriétés semblables à celles de a en déterminant le noyau de M par rapport à S ; le noyau à obtenir est un sous-ensemble d'actions qui a les propriétés suivantes :

- aucune action du noyau n'est surclassée par une autre action (stabilité interne)
- toute action hors du noyau est surclassée par au moins une action du noyau (absorption).

Il contient donc les "bonnes actions", et c'est parmi celles-ci que le décideur sera invité à faire son choix.

Le problème posé par l'utilisation du noyau est celui de son existence et de son unicité. En effet, si S contient des circuits, on n'est pas certain a priori de

l'existence d'un noyau; pour cette raison, on est obligé de réduire les actions de chaque circuit en une seule action avant de rechercher le noyau, ce qui a pour effet de supprimer une bonne partie de l'information contenue dans S .

C'est pourquoi Roy a proposé de considérer comme "bonnes actions", celles contenues dans le(s) quasi-noyau(x) de M par rapport à S .

L'existence de $\text{quasi-noyau}(x)$ est, en outre, toujours garantie.

III.4.2.2.1 Définition.

Un quasi-noyau $Q \subset X$ d'une graphe $G = (X, U)$ est un ensemble de sommets tels que :

- aucun sommet du quasi-noyau n'est joint à un autre sommet du quasi-noyau (stabilité interne)
- chaque sommet de $X \setminus Q$ est le sommet final d'au moins un chemin dont la longueur est au maximum 2 et dont le sommet initial appartient à Q (quasi-absorption).

Remarques.

1. Cette définition implique une partition de X en 3 sous-ensembles :

- Q : le quasi-noyau
- D : le sous-ensemble des "dominés" qui contient les sommets n'appartenant pas à Q et qui sont suivants d'au moins un sommet du quasi-noyau
- QD : le sous-ensemble des "quasi-dominés" contenant les sommets n'appartenant pas à Q et qui ne sont pas suivants d'un sommet du quasi-noyau.

2. Si QD est vide, alors QD est en fait un noyau.

III.4.2.2.2 Problème soulevé par l'utilisation de quasi-noyaux.

Pour un graphe G , il peut exister plusieurs quasi-noyaux; cependant, étant donné que la relation de surclassement S n'est pas nécessairement transitive, et que les sommets de QD sont des sommets qui sont surclassés par transivité par un sommet de Q , on a intérêt à choisir les quasi-noyaux auxquels sont associés le moins possible de sommets quasi-dominés.

Cette remarque nous permet de découvrir une fonction-objectif liée à la détermination des quasi-noyaux d'un graphe : c'est de chercher les quasi-noyaux minimisant le cardinal du sous-ensemble QD , appelé faiblesse du quasi-noyau, correspondant.

Le Logiciel ayant déterminé le(s) quasi-noyau(x) de faiblesse minimale de M par rapport à S , il reste à l'utilisateur à sélectionner les "bonnes actions" parmi celles des quasi-noyaux ou celles des sous-ensembles quasi-dominés correspondants.

III.4.2.2.3 Méthodes pour rechercher les quasi-noyaux [21] .

La résolution d'un programme linéaire en variables entières $z_i = \{0,1\}$ avec $z_i = 0$ si le sommet $i \in D$ ou à QD
 $z_i = 1$ s'il appartient à Q ,

pour déterminer les quasi-noyaux de faiblesse minimale étant peu efficace lorsque les dimensions du tableau du simplex sont grandes,

L'algorithme ci-dessous, utilisant les méthodes "Branch and Bound" [19] , [22] , a été développé.

III.4.2.2.3.1 Définitions préliminaires.

Soit le graphe $G = (M, U)$ associé à la relation de surclassement S , donnée par sa matrice d'incidence $A = (a_{ij})$ où $a_{ij} = 1$ s'il y a un arc entre les sommets x_i et x_j , et $a_{ij} = 0$ sinon.

Pour un sommet i , on définit :

$$\begin{aligned} S_i &= \{j \mid a_{ij} = 1\} : \text{les successeurs d'ordre un de } i \\ D_i &= \{j \neq i \mid a_{ij} = 0, \exists k : a_{ik}.a_{kj} = 1\} : \text{les} \\ &\quad \text{successeurs d'ordre 2 de } i \\ P_i &= \{j \mid a_{ji} = 1\} : \text{les prédécesseurs d'ordre un de } i \\ A_i &= \{j \neq i \mid a_{ji} = 0, \exists k : a_{jk}.a_{ki} = 1\} : \text{les pré-} \\ &\quad \text{décesseurs d'ordre deux de } i \\ p_i &= |P_i| \\ a_i &= |A_i| \\ p_i^0 &= |\{k \mid k \in P_i, z_k = 0\}| \\ p_i^1 &= |\{k \mid k \in P_i, z_k = 1\}| \\ a_i^0 &= |\{k \mid k \in A_i, z_k = 0\}| \end{aligned}$$

III.4.2.2.3.2 Principe de l'algorithme.

La relaxation choisie du problème initial consiste à résoudre un problème pour lequel la propriété de quasi-absorption est abandonnée; en résolvant ce problème relaxé, on risque donc de trouver des solutions à rejeter.

A une itération courante, un sous-problème P_k est tel qu'un certain nombre de variables z_i auront été fixées à 0 ou 1; l'objectif est alors de fixer les variables libres à 0 ou 1 en vue d'obtenir un quasi-noyau dont la faiblesse est inférieure à celle de la meilleure solution obtenue jusqu'à présent.

Pour un sous-problème P_k , est associé un ensemble I_k de solutions admissibles ou non; une solution de cet ensemble I_k est définie par :

- la valeur des variables z_i fixées à 0 ou à 1
- une combinaison particulière en 0 - 1 pour les variables z_i libres.

III.4.2.2.3.3 L'algorithme.

(a) Initialisation

- $\forall i : p_i^0 = p_i^1 = a_i^0 = 0$
- $z_{opt} = m_1$
- libérer toutes les variables

(b) test direct d'optimalité

- calculer $\underline{z} = \left\{ i \mid z_i = 0, p_i^0 = p_i \right\}$
- si $\underline{z} > z_{opt}$ go to (i)

(c) test direct d'admissibilité

- s'il existe des i tel que $z_i = 0$ et $p_i^0 + a_i^0 = p_i + a_i$
go to (i)

(d) test de résolution

- s'il existe au moins un couple (i, j) tel que z_i et z_j sont libres et $a_{ij} = 1$ et/ou $a_{ji} = 1$ go to (e)
- sinon $z_{opt} = \underline{z}$, $z_i^{opt} = z_i$ si z_i n'est pas libre
 $z_i^{opt} = 1$ si z_i est libre
- go to (i)

(e) test conditionnel d'admissibilité

- s'il existe des i tel que z_i est libre et $p_i^0 + a_i^0 = p_i + a_i$, poser $k = i$ et go to (g),
ou
- s'il existe des i tel que $z_i = 0$ et $p_i^0 + a_i^0 = p_i + a_i - 1$, chercher l'index $k \in P_i \cup A_i$ tel que $z_k \neq 0$; si z_k est libre, go to (g)

(f) test conditionnel d'optimalité

pour chaque z_i libre, calculer

$$p_i^0 = |\{k \mid k \in S_i, z_k = 0, p_k^0 = p_k^0 - 1\}| + \max(1 + p_i^0 - p_i, 0)$$

s'il existe un i tel que z_i est libre et $\underline{z} + p_i^0 >$

z_{opt} , poser $k = i$,

go to (g)

sinon, go to (h)

(g) fixation d'une variable z_k à 1

$$z_k := 1; \forall i \in S_k, p_i^1 = p_i^1 + 1$$

$\forall j \in S_k \cup P_k$, si z_j est libre :

$$- z_j := 0$$

$$- \forall h \in S_j, p_h^0 = p_h^0 + 1$$

$$- \forall h \in D_j, a_h^0 = a_h^0 + 1$$

go to (b)

(h) choix d'une variable de branchement

$\forall i$ tel que z_i est libre,

$$\text{calculer } v_i = |\{j \mid z_j \neq 1, p_j^1 = 0, a_{ji} = 1\}|$$

- sélectionner k tel que

$$v_k = \max \{v_i \mid z_i \text{ est libre}\}$$

- go to (g)

(i) régression.

Soit z_j , la dernière variable fixée; s'il n'y en a plus, les solutions $z_{opt} = (z_i^{opt})$ correspondent à des noyaux de faiblesse minimum

sinon :

- si $z_j = 0$, . libérer z_j ;
 - . $\forall h \in S_j, p_h^0 = p_h^0 - 1$
 - . $\forall h \in D_j, a_h^0 = a_h^0 - 1$
 - . go to (i)
- si $z_j = 1$, et a été fixée par un test conditionnel,
 - . libérer z_j
 - . $\forall h \in S_j, p_h^1 = p_h^1 - 1$
 - . go to (i)
- si $z_j = 1$, et a été fixée à l'étape de choix,
 - . $z_j := 0$
 - . $\forall h \in S_j, p_h^1 = p_h^1 - 1$ et $p_h^0 = p_h^0 + 1$
 - . $\forall h \in D_j, a_h^0 = a_h^0 + 1$
 - . go to (b)

III.4.2.2.3.4 Explication des étapes de l'algorithme.

- . Etape (a) : au départ, aucune variable n'est fixée; tous les sommets x_i sont placés dans QD; la faiblesse du meilleur quasi-noyau est alors : $z_{opt} = m_1$.

Les étapes (b), (c), (d) correspondent à la définition de critères de rejet sur l'ensemble I_k de solutions associé au sous-problème P_k .

- . Etape (b) : ce test, associé au critère permettant de montrer que P_k n'a pas de solution meilleure que la solution actuelle,

consiste à calculer, pour P_k , le nombre \underline{z} de sommets $i \in QD$, et à rejeter P_k si $\underline{z} > z_{opt}$ actuel.

- . Etape (c) : le test, associé au critère permettant de montrer que P_k n'a pas de solution admissible, consiste à voir s'il existe un sommet $x_i \notin Q$ et $\notin D$ tel qu'aucun de ses prédécesseurs d'ordre 2 n'appartient à Q ; si c'est le cas, la propriété de quasi-absorption est violée, et P_k est à rejeter.
- . Etape (d) : le test, associé au critère permettant de montrer qu'une solution admissible particulière est une solution optimale de P_k , consiste à voir s'il existe au moins un couple d'indices i et j tel que les sommets associés sont libres, et reliés entre eux par un ou deux arcs;
 - si c'est le cas, on ne sait pas déterminer une solution complète, puisqu'on ne peut pas dire si x_i et x_j vont appartenir à D ou à QD .
 - sinon, à partir de la solution partielle, on peut déterminer une solution admissible particulière $\tilde{x}_k \in I_k$, solution optimale de P_k tel que $\underline{z}(\tilde{x}_k) \leq \underline{z}(x)$, $\forall x \in I_k$.

En effet,

- ou bien toutes les variables sont fixées, et I_k est réduit à cette solution,
- ou bien certaines variables x_i sont libres, on les place dans Q en fixant z_i à 1; ainsi, la faiblesse de la solution complète n'augmentera pas

par rapport à celle de la solution partielle, et
 $\forall x \in I_k, \underline{z}(\tilde{x}_k) \leq \underline{z}(x)$.

Si aucun des tests précédents n'est positif, les étapes (e), (f), (g) correspondent à l'application de critères de rejet sur un sous-ensemble de I_k spécifié par une contrainte sur une variable libre x_i , c'est-à-dire en fixant z_i à 0; si l'un des critères est vérifié, on considère alors comme sous-problème courant celui constitué par le sous-problème P_k et la contrainte complémentaire sur x_i , c'est-à-dire en fixant z_i à 1.

. Etape (e) : s'il existe un i tel que :

- z_i est libre et $p_i^0 + a_i^0 = p_i + a_i$,
 alors la partie de I_k spécifiée en fixant z_i à 0 n'a pas de solution admissible.
- $z_i = 0$ et $p_i^0 + a_i^0 = p_i + a_i - 1$,
 alors il existe $k \in P_i \cup A_i$ tel que $z_k \neq 0$:
 ou bien z_k est libre; si on fixe z_k à 0, alors la partie de I_k spécifiée en fixant z_i à 0 ne possède pas de solution admissible,
 ou bien $z_k = 1$, et on ne sait pas dire si ce sous-ensemble contient des solutions admissibles ou non.

. Etape (f) : on cherche un sommet x_i libre tel que
 $\underline{z} + f_i^0 > z_{opt}$ actuel où f_i^0 est une pénalité; si on trouve un tel sommet, on peut montrer que le sous-ensemble de I_k déterminé par la fixation de z_i à 0 donne pour chaque solution de ce sous-ensemble une valeur \underline{z} supérieure au z_{opt} actuel.

- . Etape (g) : elle correspond à la résolution de la relaxation du sous-problème P_k courant obtenu :
 - par séparation, d'après le principe du choix de la variable de branchement; pour que cette variable n'appartienne pas à QD , on la fixera à 1
 - parce que l'un des tests (e) ou (f) ou (g) est positif.
Par conséquent, il faut :
 - fixer la variable z_k à 1
 - mettre à jour les variables π_i^0, π_i^1, \dots utilisées
 - imposer que la contrainte de stabilité soit vérifiée.
- . Etape (h) : on choisit la variable libre qui assurera que le plus grand nombre de sommets qui n'appartiennent pas encore à Q ou à D y appartiennent.
- . Etape (i) : on parcourt une des branches de l'arborescence jusqu'à un sommet-associé à un sous-problème P_k - pour lequel l'ensemble des solutions correspondantes répond positivement à un des tests (b), (c) ou (d), puis on revient sur ses pas jusqu'au premier sommet dont un des suivants correspond à un ensemble non-encore testé. On arrête quand tous les sommets, ou sous-problèmes, ont été explorés, c'est-à-dire quand toutes les variables fixées ont été examinées; elles le sont de la droite vers la gauche.
Soit z_k , la dernière variable fixée examinée :

- si elle égale 1 :
 - ou bien elle a été fixée par un test conditionnel; par conséquent, les deux sous-problèmes ont été examinés et :
 - on libère z_k ,
 - on met à jour certaines variables utilisées,
 - on examine une autre variable fixée,
 - ou bien elle a été fixée après avoir choisi une variable de branchement, il reste donc à examiner le sous-problème associé à $z_k = 0$
- si elle égale 0, elle a été fixée soit pour examiner le sous-problème associé à $z_k = 0$, soit pour respecter la contrainte de stabilité (étape (g)); par conséquent :
 - on libère z_k
 - on examine une autre variable fixée.

Remarque : les paramètres pouvant faire l'objet d'une analyse de sensibilité sont :

- le seuil de concordance
- les seuils de discordance
- le test de concordance.

III.4.3 La méthode Electre II [20] , [23] .

III.4.3.1 Objectif.

Electre II classe les actions suivant des préordres partitionnés en une suite ordonnée de classes d'indifférence globale allant du "meilleur" au "pire". Ces classements sont obtenus à partir de deux relations de surclassement correspondant à deux niveaux de risques différents pour chacun des filtres définis au point III.4.1; l'une traduisant un surclassement fort et l'autre un surclassement faible.

La méthode Electre II procède en 3 temps :

1. construction de relations de surclassement déterministes,
2. construction de 2 préordres totaux à partir des relations de surclassement,
3. comparaison des 2 préordres obtenus et élaboration d'un rangement final.

III.4.3.2 Principe de construction des relations de surclassement.

Pour établir les 2 relations, le décideur doit rentrer les paramètres suivants :

1. trois seuils de concordance correspondant à des niveaux d'accords exigés distincts, $C1, C2, C3$, tel que :
 $1 > C1 > C2 > C3 > 0$
2. deux seuils de discordance, pour chaque critère i , correspondant à des intensités de désaccords acceptés différents, tel que :

$$0 < d_{1i} < d_{2i} \quad \forall i.$$

On considère chaque couple d'actions (a,b);

étant donné que :

$I^+(a,b)$, $I^-(a,b)$, $I^{\bar{}}(a,b)$, $P^+(a,b)$, $P^-(a,b)$, $P^{\bar{}}(a,b)$ et $C(a,b)$ ont la même définition que pour Electre I,

on dit que a surclasse fortement b si :

$$P^+(a,b) \geq P^-(a,b) \text{ et}$$

$$(i) \text{ ou bien } C(a,b) \geq C_1, \text{ et } X_i(b) - X_i(a) \leq d_{2i}, \forall i \in I^-(a,b)$$

$$(ii) \text{ ou bien } C(a,b) \geq C_2, \text{ et } X_i(b) - X_i(a) \leq d_{1i}, \forall i \in I^-(a,b)$$

(i) et (ii) définissent 2 options pour le surclassement fort : ou bien on est plus exigeant sur la concordance, ou bien on est plus exigeant sur la discordance;

on dit, de plus, que a surclasse faiblement b si :

$$P^+(a,b) \geq P^-(a,b) \text{ et}$$

$$C(a,b) \geq C_3, \text{ et } X_i(b) - X_i(a) \leq d_{2i}, \forall i \in I^-(a,b);$$

on est donc moins exigeant simultanément sur la concordance et les discordances.

III.4.3.3 Principe de la détermination des classements.

Si on considère une action a,

1. plus elle est surclassée fortement par un grand nombre d'actions, et plus elle sera mal classée; c'est l'optique du classement direct, dans lequel l'action a précède l'action b si, et seulement si, le nombre d'actions surclassant fortement a est inférieur au nombre d'actions surclassant fortement b;
2. plus elle surclasse fortement un grand nombre d'actions, et plus elle devrait être bien classée :

c'est l'optique du classement indirect, dans lequel l'action a sera précédée par l'action b si, et seulement si, le nombre d'actions surclassées fortement par a est inférieur au nombre d'actions surclassées fortement par b.

Dans les 2 classements, la relation de surclassement faible permet de départager les ex-aequo.

III.4.3.4 Comparaison des 2 préordres obtenus.

Si les deux classements ne sont pas trop distincts - en terme du coefficient de corrélation de Spearman [15], par exemple -, le classement médian, que l'on obtient en affectant à chaque action la moyenne des rangs qu'elle avait dans les classements directs et indirects, peut être retenu comme classement final par l'utilisateur; sinon, il devrait réviser les évaluations qu'il a données aux actions.

Remarques :

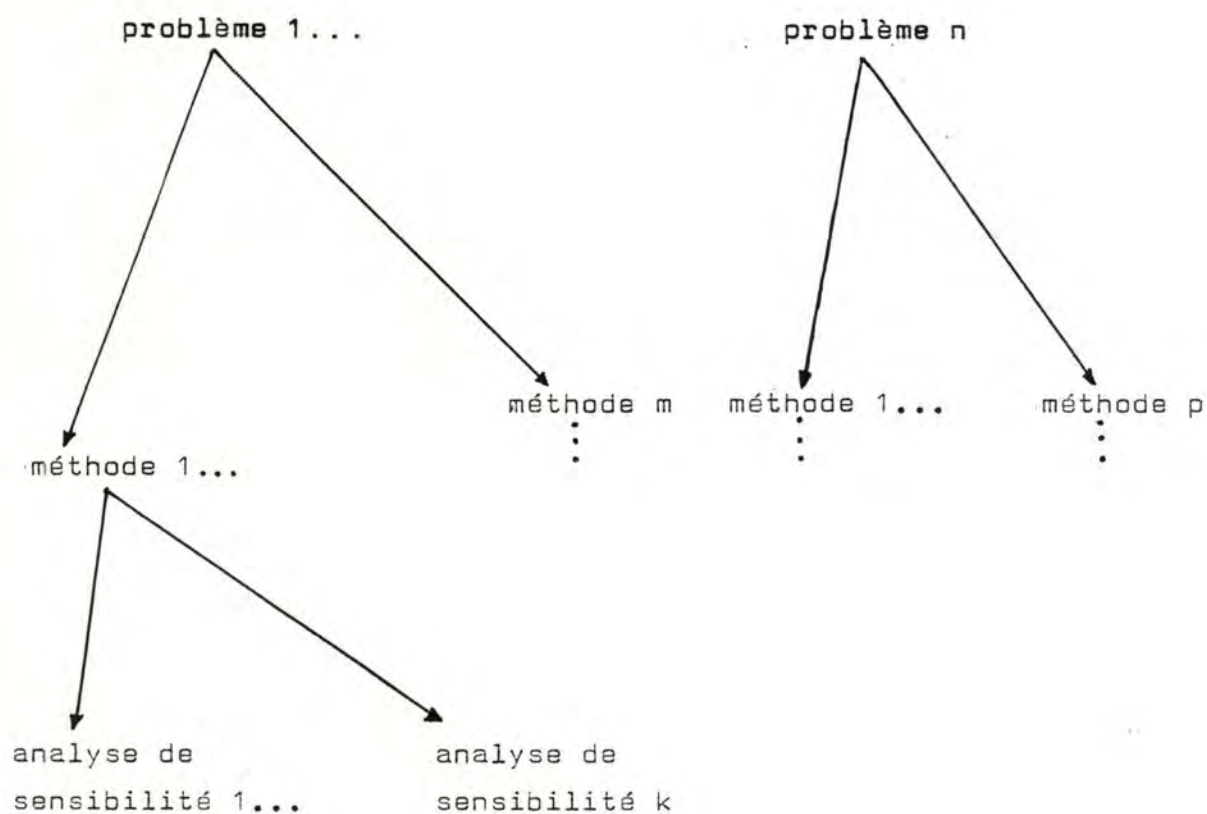
1. les paramètres pouvant faire l'objet d'une analyse de sensibilité sont :
 - les seuils de concordance,
 - les seuils de discordance,
 - le test de concordance.
2. Les autres méthodes multicritères, et en particulier la méthode Electre III sont expliquées en annexe.
3. Les divers classements définis ci-dessus sont en fait ceux proposés par H. Thiriez (Initiation au calcul économique, Dunot, 1977).

CHAPITRE IV.

Chapitre IV. L'implémentation du logiciel.

Résumons ce que nous avons dit au point 1.2.2 : le logiciel conçu ne se contente pas d'exécuter des méthodes multicritères sur un problème, auquel sont associées des données initiales, en vue d'obtenir des résultats; il permet en plus l'enregistrement, l'impression et la suppression des données initiales et des résultats obtenus.

Les possibilités du logiciel du point de vue des exécutions, des impressions, et des suppressions ont été établies à partir des considérations suivantes, schématisées par la figure 1 :



- figure 1 -

1. Le logiciel peut traiter plusieurs problèmes distincts; nous avons fixé une limite théorique de 89 problèmes multicritères qui peuvent être associés simultanément au logiciel.
2. Chaque problème peut être "résolu" par différentes méthodes multicritères.
3. Pour une méthode donnée, chaque problème peut faire l'objet de plusieurs analyses de sensibilité; une limite théorique de 99 analyses de sensibilité associées simultanément à une méthode pour un problème donné a été fixée.
De plus, à chaque problème est associé un certain nombre de critères et un certain nombre d'actions; nous avons limité arbitrairement la possibilité d'utiliser le logiciel à des problèmes multicritères ne dépassant pas 50 actions et 50 critères.

En vue d'assurer sa portabilité, le logiciel a été programmé en fortran 77 [30] ; il a été conçu de manière à ce qu'il soit facilement utilisable.

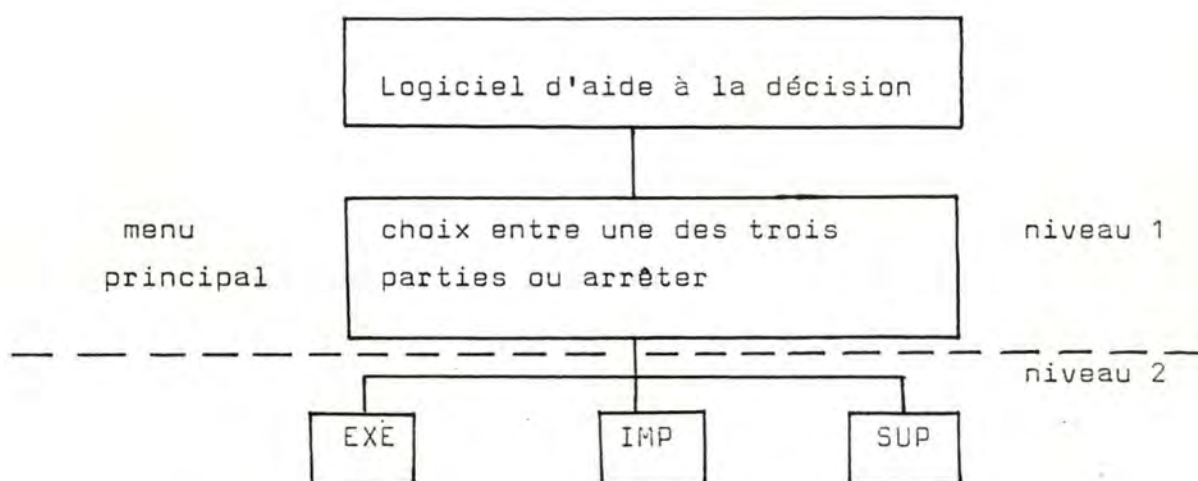
La structure arborescente du programme comprend trois parties correspondant aux possibilités du logiciel :

- exécution (EXE)
- impression (IMP)
- suppression (SUP)

Nous décrivons dans la suite les différentes parties du programme.

La structure arborescente du programme se base sur une décomposition en niveaux sous forme de menus qui guident l'utilisateur.

Le menu principal permet le choix entre EXE, IMP, SUP ou ARR (où ARR signifie l'arrêt du programme. En fonction de ce choix, on branchera vers une des possibilités proposées.



IV.1 Exécution.

Le menu d'exécution propose à l'utilisateur le choix entre une méthode multicritère et le retour au menu principal. Ces méthodes sont :

- Electre I (ELI).
- Electre II (ELII).
- Electre III (ELIII).
- Oreste (OR).
- Saaty (SATY)
- rangement médian (RGMD).
- MAL (MAL).
- UTA I (UTAI).
- UTA II (UTAI).

IV.1.1 La forme du menu-exécution.

Le menu-exécution est élaboré d'après les contraintes suivantes :

1. L'utilisateur peut désirer exécuter une méthode multicritère sur un problème déjà enregistré, et auquel sont donc déjà associées une ou plusieurs analyses de sensibilité réalisées sur une ou plusieurs méthodes; si c'est le cas, il peut :
 - vouloir exécuter cette méthode à partir d'une analyse de sensibilité enregistrée associée à la même méthode, en modifiant certains paramètres ou certains rapports de préférence;
 - vouloir exécuter cette méthode à partir de k_1 rapports de préférence partielle entre les m_1 actions enregistrés, et associés à une analyse de sensibilité d'une méthode quelconque, qu'il pourra éventuellement modifier.
2. La méthode SATY (cfr II.2) n'admet comme rapports de préférence partielle entre les actions et comme rapports de préférence entre les critères que ceux établis à partir de la méthode de rangement par des comparaisons par paires de Saaty (cfr II.1.2).
3. L'utilisateur, lorsqu'il désire utiliser une des 2 méthodes UTA sur un problème multicritère déjà enregistré, doit pouvoir demander l'exécution d'une des 3 étapes définies en II.3.1, c'est-à-dire :
 - soit une analyse de sensibilité (étape 1),
 - soit une analyse post-optimale (étape 2),
 - soit une relation de surclassement (étape 3).

4. La méthode UTA I n'utilise pas de rapports de préférence partielle entre les actions comme données de départ.

Le schéma du menu-exécution est alors le suivant :

1. si l'utilisateur désire exécuter une méthode multicritère :
2. s'il existe un ou plusieurs problèmes enregistrés, et si l'utilisateur désire exécuter une méthode sur un de ces problèmes :
 - il sélectionne un de ceux-ci,
 - il sélectionne la méthode multicritère,
 - si cette méthode est une méthode UTA, et si une ou plusieurs analyses de sensibilité lui sont associées pour le problème sélectionné, il choisit l'une des trois étapes,
 - si la méthode choisie est UTAPII et si l'utilisateur désire faire une analyse de sensibilité (étape 1), ou si la méthode n'est pas une méthode UTA, alors :
3. si à cette méthode lui sont associées une ou plusieurs analyses de sensibilité pour le problème sélectionné, il peut soit :
 - indiquer qu'il désire exécuter la méthode choisie à partir d'une de ces analyses de sensibilité;
 - utiliser comme rapports de préférence partielle entre les actions ceux associés à une analyse de sensibilité d'une méthode quelconque;
 - ne choisir aucune des 2 possibilités précédentes, car il estime que les paramètres et les rapports de préférence qu'il a à rentrer sont sensiblement

différents de ceux déjà enregistrés.

Sinon, il peut :

- utiliser comme rapports de préférence partielle entre les actions ceux associés à une analyse de sensibilité d'une méthode quelconque;
- refuser cette possibilité, car les paramètres et les rapports de préférence qu'il à rentrer sont sensiblement différents de ceux déjà enregistrés.

2. Sinon, l'utilisateur indique la méthode multicritère qu'il désire appliquer sur le nouveau problème multicritère.

1. Sinon, retour au menu principal.

Remarque : avant de choisir une méthode multicritère, l'utilisateur peut obtenir pour chacune des méthodes implémentées, une information succincte sur son objectif, les paramètres à rentrer et les résultats qu'elle fournit; cette possibilité est référencée sous le nom de DOC.

IV.1.2 La forme des traitements.

Le squelette de la routine principale associée à chaque méthode est similaire.

L'origine des traitements associés à cette routine trouve sa justification dans les considérations suivantes :

1. Pour chacune des méthodes, l'utilisateur doit rentrer plusieurs des trois types de rapports de préférence définis au chapitre I;

- 1.1 la routine "evalac" permet la saisie et l'enregistrement des rapports de préférence partielle entre les actions pour un critère donné; le menu associé à cette routine permet à l'utilisateur le choix d'une des 5 possibilités :
- la construction d'une relation de préférence quelconque;
 - la construction d'une fonction d'utilité (1);
 - l'utilisation de la méthode de rangement par des comparaisons par paires (2);
 - s'il dispose du préordre total sur les actions exigé, la saisie du préordre à l'aide de nombres traduisant ce classement (3);
 - obtenir des informations succinctes sur l'objectif et la façon d'utiliser chacune des possibilités ci-dessus, avant de choisir l'une d'elles.
- 1.2 La routine "estpdc" permet la saisie et l'enregistrement des rapports de préférence entre les critères; le menu associé à cette routine permet à l'utilisateur le choix d'une des 4 possibilités :
- la construction d'une relation de préférence quelconque;
 - l'utilisation de la méthode de rangement par des comparaisons par paires;
 - s'il dispose du préordre total exigé, la saisie du préordre à l'aide de nombres traduisant ce classement;
 - obtenir des informations succinctes sur l'objectif et la façon d'utiliser chacune des possibilités ci-dessus, avant de choisir

l'une d'elles.

1.3 La routine "evlglo" permet la saisie et l'enregistrement des rapports de préférence globale entre les actions; le menu associé à la routine permet à l'utilisateur le choix d'une des 4 possibilités :

- la construction d'une relation de préférence quelconque;
- l'utilisation de la méthode de rangement par des comparaisons par paires;
- s'il dispose du préordre total exigé, la saisie du préordre à l'aide de nombres traduisant ce classement;
- obtenir des informations succinctes sur l'objectif et la façon d'utiliser chacune des possibilités ci-dessus, avant de choisir l'une d'elles.

Ainsi :

- Les méthodes ELI, ELII, ELIII, OR utilisent "Evalac" et "Estpdc".
- La méthode UTAPII utilise "Evalac" et "Evl glo".
- La méthode UTAPI utilise "Evl glo".
- La méthode SATY ne peut utiliser que la routine correspondant à l'utilisation de la méthode de rangement par des comparaisons par paires.
- La méthode MAL utilise "estpdc". Pour l'obtention des rapports de préférence partielle entre les actions, elle peut utiliser les routines correspondant aux possibilités (1), (2) et (3).
- La méthode RGMD utilise "evalac". Pour l'obtention des rapports de préférence entre les critères, RGMD utilise

une routine "saiswg", qui en fonction de l'expression que l'utilisateur désire donner de ces rapports (cfr annexe II.4.2), permet **soit** :

- d'utiliser "estpdc", si l'utilisateur désire donner des poids précis à chaque critère;
- de saisir la position de chaque critère sur une échelle ordinale, si l'utilisateur désire obtenir toutes les médianes associées à chacun des systèmes de poids (p_1, \dots, p_k) respectant la position de chacun des critères sur cette échelle.

2. Hormis pour les méthodes SATY et RGMD, l'utilisateur doit rentrer des paramètres tels qu'ils sont définis dans la présentation des différentes méthodes. Pour saisir la valeur de ces paramètres, sont donc associées des routines de saisie de données.
3. Lorsque l'utilisateur a sélectionné une analyse de sensibilité, ou lorsqu'il désire en réaliser une supplémentaire, il est nécessaire d'organiser la saisie des paramètres ou des rapports de préférence qu'il veut modifier : c'est le premier objectif d'une routine, associée à chaque méthode, que nous appellerons "SEN". Avant d'exécuter une analyse de sensibilité, il est nécessaire de s'assurer que l'analyse de sensibilité demandée n'a pas déjà été exécutée et est encore enregistrée. C'est le second objectif de la routine "SEN".

IV.1.2.1 Le squelette de la routine principale associée aux méthodes d'agrégation.

Compte tenu des considérations sur la forme du menu-exécution et sur la forme des traitements, le squelet-

te de la routine associée à chaque méthode d'agrégation est le suivant :

1. s'il s'agit d'un nouveau problème multicritère, ou d'un problème déjà enregistré mais pour lequel l'utilisateur désire rentrer de nouveaux rapports de préférence partielle entre les actions, saisie et enregistrement des rapports de préférence partielle entre les actions en respectant la considération 1 du point précédent.
2. - S'il s'agit d'un nouveau problème, ou d'un ancien problème pour lequel on n'a pas sélectionné une analyse de sensibilité enregistrée :
 - saisie des paramètres associés à la méthode compte tenu de la considération 2 du point précédent.
 - saisie et enregistrement des rapports de préférence entre les critères suivant la considération 1 du point précédent.
- Par contre, si l'utilisateur a sélectionné une analyse de sensibilité enregistrée, le logiciel "va chercher" les paramètres et les rapports de préférence entre critères associés à celle-ci.
3. Pour un problème déjà enregistré, vérification que l'analyse de sensibilité demandée n'est pas déjà enregistrée.
4. S'il s'agit d'un nouveau problème, ou si l'analyse de sensibilité demandée n'est pas encore enregistrée :
appel de la routine d'exécution de la méthode.
5. Retour au menu d'exécution.

IV.1.2.2 Le squelette de la routine principale associée aux méthodes UTA.

Le choix entre l'exécution d'une des 3 étapes par l'utilisateur modifie légèrement le squelette de la routine principale que voici :

1. Pour la méthode UTAI :

s'il s'agit d'un nouveau problème, ou d'un problème déjà enregistré pour lequel l'utilisateur désire effectuer une analyse de sensibilité (étape 1) à partir de nouveaux rapports de préférence partielle entre les actions, saisie et enregistrement des rapports de préférence partielle entre les actions en respectant la considération 1 du point IV.1.2.

2. - S'il s'agit d'un nouveau problème, ou d'un problème déjà enregistré pour lequel l'utilisateur n'a pas sélectionné une analyse de sensibilité enregistrée :

- saisie des paramètres associés à UTAI ou à UTAI.
- saisie et enregistrement des rapports de préférence globale entre actions suivant la considération 1 du point IV.1.2.

- par contre, si l'utilisateur a sélectionné une analyse de sensibilité enregistrée pour réaliser l'étape 1 ou l'étape 2, le logiciel "va chercher" les paramètres et les rapports de préférence globale entre actions associés à celle-ci.

3. Pour un problème déjà enregistré, et si l'utilisateur désire réaliser l'étape 1, vérification que l'analyse

de sensibilité demandée n'est pas déjà enregistrée.

4. - S'il s'agit d'un nouveau problème, ou si l'analyse de sensibilité demandée n'est pas encore enregistrée et si l'utilisateur désire réaliser l'étape 1 : appel de la routine d'exécution de la méthode.
 - S'il désire réaliser l'étape 2 :
entrée secondaire dans la routine d'exécution de la méthode pour réaliser l'analyse post-optimale.
 - S'il désire réaliser l'étape 3 :
appel de la routine correspondant à la construction de la relation de surclassement.

5. Retour au menu d'exécution.

IV.1.2.3 Description de la routine d'exécution des méthodes.

Après l'exécution de l'analyse de sensibilité demandée, l'utilisateur doit avoir la possibilité de réaliser d'autres analyses de sensibilité pour la même méthode multicritère en modifiant certains paramètres ou certains rapports de préférence; la routine SEN sera utilisée pour réaliser ces modifications et vérifier que la nouvelle analyse de sensibilité demandée n'est pas enregistrée.

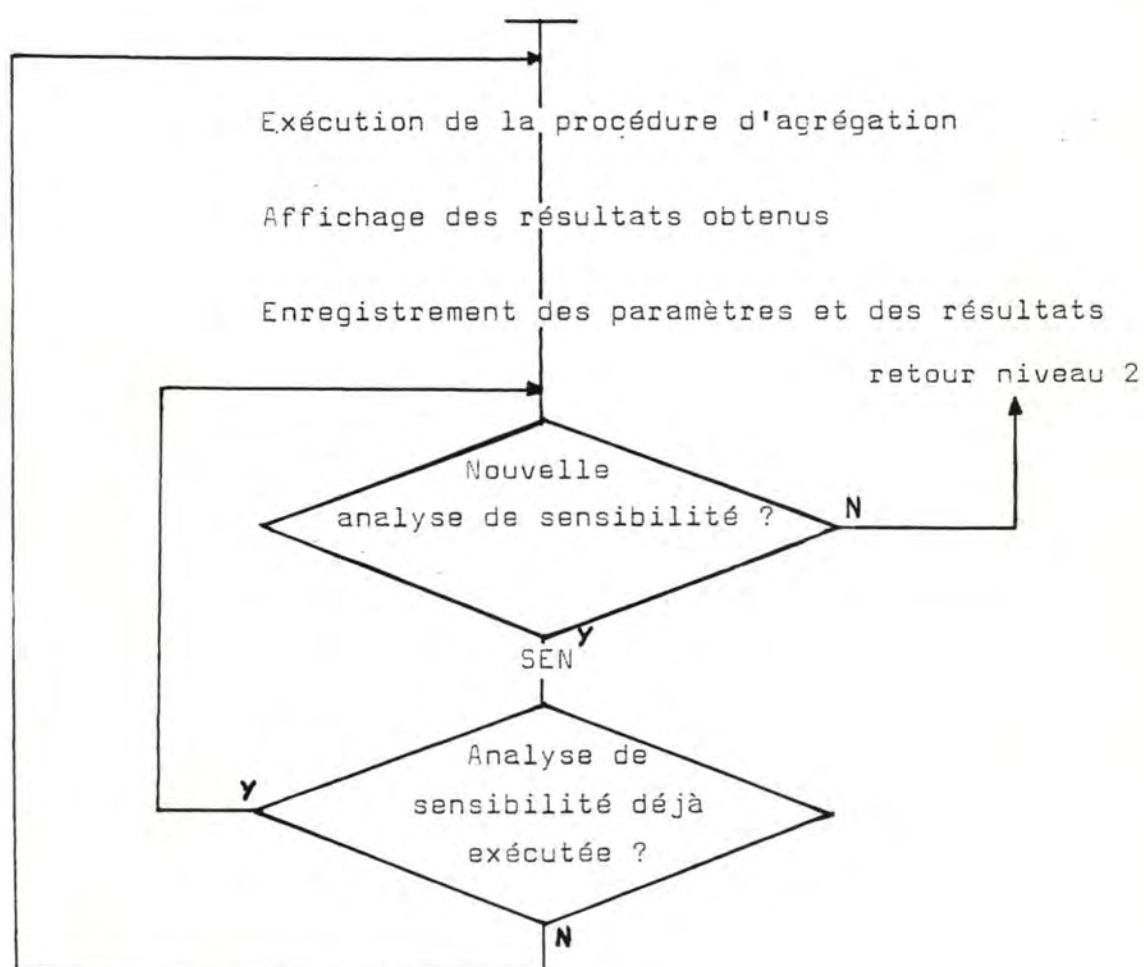
Pour les méthodes UTA, l'utilisateur devra de plus pouvoir exécuter plusieurs analyses post-optimales (étape 2) ou construire plusieurs relations de surclassement (étape 3); à chacune de ces 2 étapes **seront** donc associées des routines, appelées respectivement SOPT et SRCL, qui

auront des objectifs similaires à SEN.

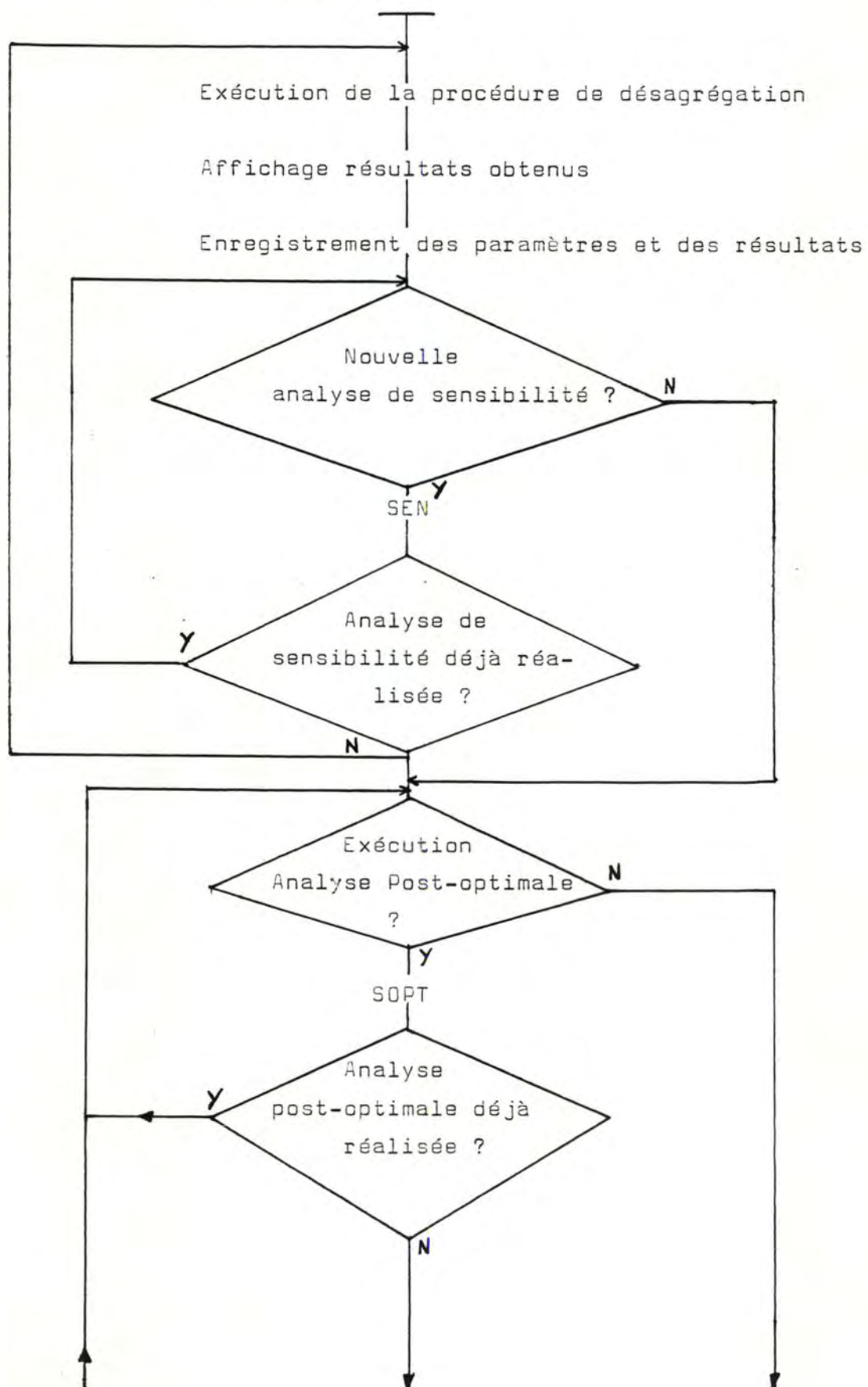
SOPT assurera la saisie des paramètres nécessaires à la réalisation de l'étape II et vérifiera qu'une telle analyse post-optimale n'est pas déjà réalisée.

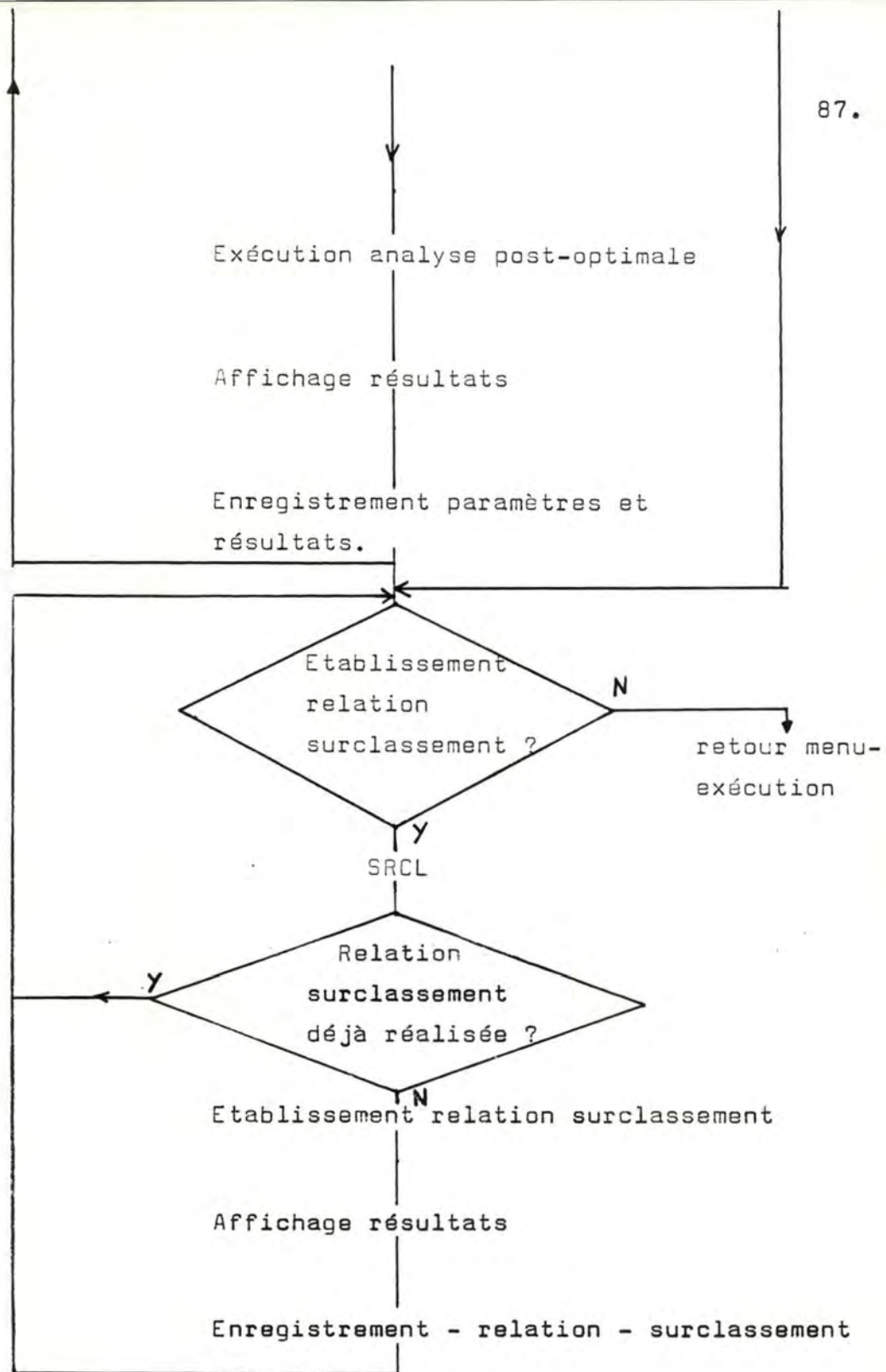
SRCL permettra à l'utilisateur de dire quelles fonctions d'utilités il désire retenir pour construire la relation de surclassement; SRCL vérifiera ensuite que cette étape 3 n'est pas déjà enregistrée.

IV.1.2.3.1 Schéma de la routine d'exécution associée aux méthodes d'agrégation.



IV.1.2.3.2 Schéma de la routine d'exécution associée
aux méthodes UTA.





IV.1.2.3.3 Précisions sur l'enregistrement des données et des résultats.

L'enregistrement des rapports de préférence est réalisé de telle sorte que non seulement sont enregistrés les préordres utilisés par les méthodes multicritères, mais aussi les rapports de préférence tel que l'utilisateur aura pu les exprimer en utilisant les outils développés au chapitre II; ainsi :

- si l'utilisateur construit une relation de préférence quelconque, le logiciel enregistrera cette relation.
- Si la méthode des rangements par des comparaisons par paires de Saaty est utilisée, sont enregistrées :
 - la matrice des comparaisons par paires,
 - la valeur propre dominante et le vecteur propre associé.
- Si une fonction d'utilité est construite, on enregistrera les points déterminés de cette fonction d'utilité, ainsi que le préordre sur les actions déduit de la fonction d'utilité.

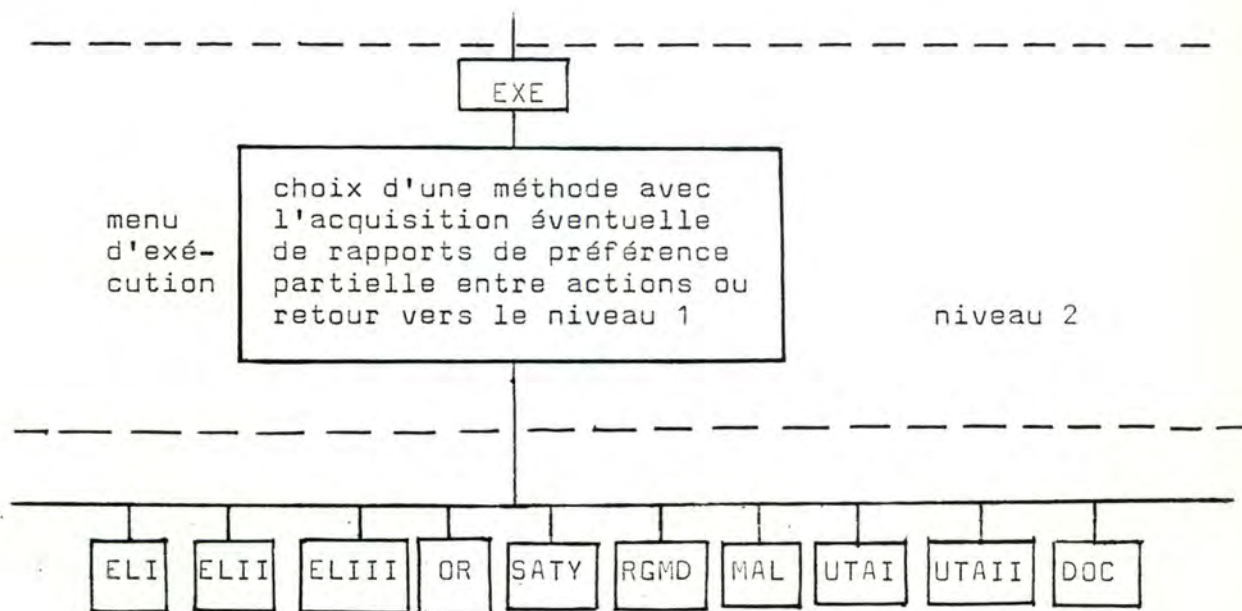
Outre les rapports de préférence d'un problème qui peuvent être utilisés par plusieurs méthodes, on enregistrera les paramètres et les résultats qui sont des données propres à chaque analyse de sensibilité et à chaque méthode.

Ces informations sont enregistrées sur des fichiers à accès direct.

On distinguera donc les fichiers, contenant les rapports de préférence, qui sont communs à plusieurs méthodes, et

les fichiers, propres à chaque méthode, qui contiendront les paramètres et les résultats associés à l'exécution de méthodes multicritères. L'ensemble des fichiers utilisés est décrit en annexe III.

Résumons la structure de la partie exécution :



niveau 3 : routine principale associée à chaque méthode ou information procurée par **DOC**.

IV.2 Impression.

Cette partie doit permettre :

1. une exploitation meilleure et plus facile des résultats apportés par les méthodes multicritères.
2. une remise en question aisée des rapports de préférence et des paramètres choisis pour une analyse de sensibilité, pour pouvoir éventuellement en choisir de nouveaux.

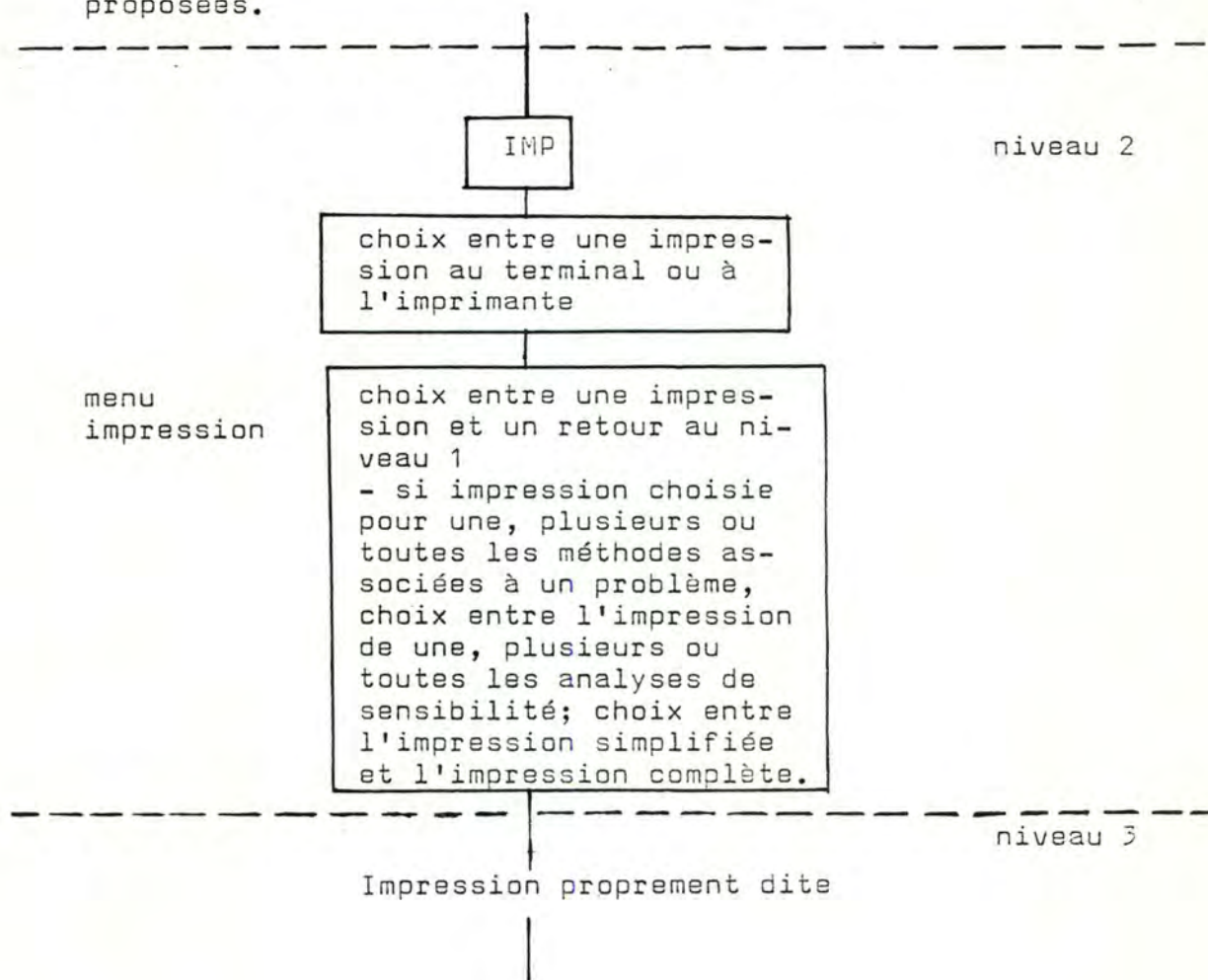
Les différents types d'impression sont déterminés par les contraintes suivantes :

1. Pour un problème donné, l'utilisateur doit pouvoir demander, pour une ou plusieurs méthodes, l'impression des données rentrées par l'utilisateur et des résultats de **une ou plusieurs** analyses de sensibilité.
2. Pour une analyse de sensibilité donnée, l'utilisateur peut vouloir :
 - obtenir uniquement l'impression des rapports de préférence tels qu'ils sont utilisés par une méthode donnée, c'est-à-dire sous forme de préordre :
c'est l'impression simplifiée.
 - savoir **comment** ces préordres ont été déterminés à partir de l'impression des relations de préférence initiale, des matrices des comparaisons par paires ou des fonctions d'utilité

qui auraient pu être utilisées pour déterminer ces préordres : c'est l'impression complète.

3. Les impressions doivent pouvoir se faire à partir d'un terminal papier ou de l'imprimante rapide de l'ordinateur.
4. Le nombre de critères et le nombre d'actions n'étant pas fixés a priori pour un problème multicritère, on est obligé de rejeter l'impression par tableau où à une colonne correspondrait une action ou un critère; ces impressions associeront donc une action ou un critère à une ligne.

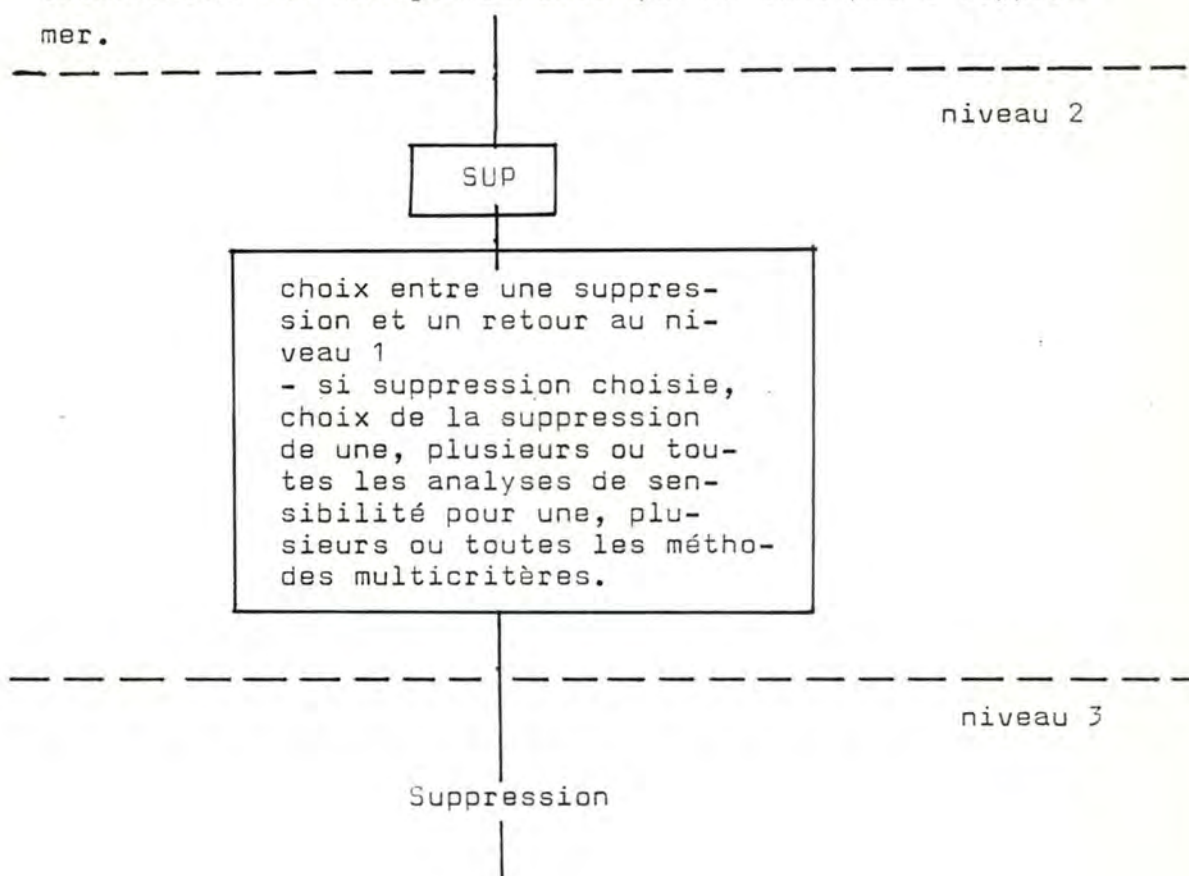
Une structure arborescente guide l'utilisateur et facilite une exploitation efficace des possibilités d'impressions proposées.



IV.3 Suppression.

L'objectif est de supprimer pour un problème multicritère donné, une, plusieurs ou toutes les analyses de sensibilité, qui n'ont plus d'intérêt, associées à une, plusieurs ou toutes les méthodes.

Ces suppresssions se font par recopie sur un fichier intermédiaire des enregistrements qui ne sont pas à supprimer.



CHAPITRE V.

Chapitre V : Expériences pratiques.

Ce chapitre montre l'utilisation qui peut être faite de certaines méthodes multicritères dans un domaine où l'on aurait peut-être pas pensé a priori les utiliser : celui de l'aide à l'identification des entérobactéries.

VI. Le problème envisagé : l'identification des entérobactéries.

Le personnel des laboratoires de biologie clinique se trouve fréquemment confronté au problème d'identifier, à partir de souches (selles ou urine, par exemple) prélevées sur un individu présumé malade, la bactérie présente dans cette souche.

Réaliser cette identification revient à choisir parmi plusieurs bactéries potentielles, celle(s) qui ressemble(nt) le plus, ou suffisamment, au germe inconnu prélevé chez l'individu présumé malade. Par bactérie potentielle, il faut entendre ici une bactérie qui pourrait être le germe inconnu et pour laquelle des spécialistes de la microbiologie ont déterminé un ensemble de caractéristiques qui suffit, au moins en principe, à la reconnaître.

A partir de ces caractéristiques, certaines firmes pharmaceutiques ont établi, pour les bactéries de la famille des entérobactéries, et proposé aux laboratoires de biologie des systèmes de microtests prêts à l'emploi et qui devraient permettre au personnel une identification biochimique aisée des bactéries de cette famille.

Les méthodes d'identification associées à ces systèmes s'appuient sur le fait que des souches correspondant à

une même entérobactérie présentent un ensemble prévisible de réponses - positives ou négatives - à une série de tests biologiques, c'est ainsi que différentes firmes pharmaceutiques ont établi leur tableau de pourcentages reprenant la fréquence espérée de réactions positives pour chaque test et pour chaque espèce d'entérobactérie; cette fréquence étant calculée en déterminant le rapport du nombre N_j de réactions positives au test t_j obtenues sur un nombre N "élevé" de souches bien identifiées d'une bactérie donnée. On peut citer comme procédures d'identification traditionnelles [6], celles utilisant les arbres de décision, ainsi que celles utilisant des probabilités conditionnelles; elles prennent appui sur ce qu'on appelle la forme testable du germe inconnu u : $X(u) = (X_1(u), \dots, X_n(u))$ où $X_j(u) = 1$, si u a occasionné une réaction positive au test t_j et $X_j(u) = 0$ dans le cas contraire.

J. Fichefet et J.P. Leclercq ont montré [6] l'intérêt de l'utilisation de méthodes multicritères pour permettre l'identification des entérobactéries; on peut, en effet, considérer le processus d'identification comme un processus d'aide à la décision multicritère dans lequel les entérobactéries jouent le rôle d'actions et les tests biochimiques le rôle de critères.

Les rapports de préférence partielle entre les actions pour un critère j nécessaires peuvent être obtenus à partir de la forme testable $X(u)$ et du tableau de pourcentages. Ainsi l'évaluation $Z_j(u, b_i)$ de la bactérie b_i pour le critère j est déterminée à partir de l'élément $Y_j(u, b_i)$ de ce tableau et vaut :

$$Z_j(u, b_i) = \begin{cases} 100X_j(b_i) & \text{si } X_j(u) = 1, \\ 100 - 100 X_j(b_i) & \text{si } X_j(u) = 0. \end{cases} \quad (\text{formule 1})$$

Disposant des évaluations $Z_j(u, b_i)$, on peut appliquer une méthode d'agrégation, par exemple Electre II; on choisit alors comme identification du germe u , la bactérie qui, par exemple, se trouve la mieux classée par rapport aux classements déterminés par cette méthode.

A partir de 200 formes testables pour lesquelles l'identification exacte était en principe disponible, les recherches effectuées à l'Institut d'Informatique ont permis de plus :

1. de déterminer pour quelle(s) valeur(s) de ses paramètres, chaque méthode multicritère utilisée donne les meilleures identifications;
2. de comparer les résultats donnés par différentes méthodes, et de déterminer celle(s) qui fournisse(nt) les meilleures identifications.

Remarques :

1. Les recherches effectuées ont notamment mis en évidence la nécessité de transformer les évaluations initiales données par la formule 1; en effet, pour un test t_j , si les évaluations données par la formule 1, pour deux actions a et b , sont $Y_j(u, a) = 90$ et $Y_j(u, b) = 95$, il n'y a vraisemblablement pas suffisamment de bonnes raisons de préférer b à a par rapport à ce test. Des formules de transformation ont été adoptées; les évaluations $Z_j(u, b_i)$ finalement retenues valent alors, $\forall i, j$:

$$\left[\begin{array}{ll} 100 & \text{si } Y_j(u, b_i) \geq 70 \\ 50 & \text{si } 30 \leq Y_j(u, b_i) < 70 \\ Y_j(u, b_i) & \text{si } Y_j(u, b_i) < 30 \end{array} \right.$$

cette stratégie est appelée "stratégie 70"

ou

$$\left[\begin{array}{ll} 100 & \text{si } Y_j(u, b_i) \geq 50 \\ Y_j(u, b_i) & \text{sinon} \end{array} \right.$$

si la "stratégie 50" est adoptée.

2. La méthode HERMES [7] a permis de déterminer de façon précise l'importance relative de chaque test dans le processus d'identification des entérobactéries; nous avons utilisé ces poids pour réaliser les expérimentations.

V.1.1 Les expérimentations réalisées.

Les expérimentations que nous avons effectuées portent sur 15 formes testables différentes extraites d'un rapport de contrôle de qualité concernant 57 laboratoires ordinaires de biologie clinique. Des laboratoires d'expertise ont établi que ces 15 profils correspondent à des souches d'une même entérobactérie (*Klebsiella ozaenae*). Il est donc clair que des erreurs techniques de manipulation peuvent être commises en laboratoire ordinaire.

Sans pouvoir tirer des conclusions telles que celles issues des recherches effectuées à l'Institut, le nombre de profils testés est trop peu élevé pour cela, les tests effectués permettent cependant :

1. de montrer de façon précise comment le processus d'identification est réalisé pour chacune des deux méthodes utilisées :
ELECTRE III et ORESTE,
2. de tirer quelques enseignements à la lumière des différentes analyses de sensibilité effectuées.

V.1.1.1 L'expérimentation vis-à-vis de la méthode Electre III.

L'un des avantages de cette méthode - présentée en annexe II - réside dans la possibilité de définir pour chaque critère un seuil de préférence sp , ainsi qu'un seuil d'indifférence si . Si l'on estime, par exemple, que deux actions a et b , dont la différence des évaluations pour un critère i , $X_i(a) - X_i(b) \geq 0$, ne dépasse pas 10, doivent être considérées comme indifférentes pour ce critère, on introduira un seuil d'indifférence de 10; tandis qu'un seuil de préférence fixé à 15 signifie que l'on considère a préféré à b pour le critère i dès que la différence $X_i(a) - X_i(b)$ est supérieure à 15; au vu de la remarque 1 du point V.1, on pourra ainsi utiliser les évaluations initiales définies par la formule 1 sans appliquer de formules de transformation.

On doit, de plus, introduire un seuil de veto sv - dont la signification est similaire aux seuils de discordance associés à Electre I et à Electre II - qui permettra de rejeter le surclassement de a sur b dès que $X_i(b) - X_i(a)$

$\geq sv$. Remarquons que les recherches effectuées à l'Institut ont mis en évidence le fait que des seuils de discordance élevés produisaient de meilleurs résultats.

Nous avons convenu que la bactérie proposée comme identification du germe inconnu par Electre III est celle à laquelle est associé le rang le plus élevé dans le classement médian. Si r désigne le rang associé à *Klebsiella ozaenae*, l'identification proposée par Electre III est : correcte (C) si ce rang est supérieur à tous les autres rangs, suspecte (X) si plusieurs bactéries dont *Klebsiella ozaenae* ont le rang maximum, erronée (E) dès que r est strictement inférieur au rang attribué à d'autres bactéries.

Remarquons que les classements déterminés par Electre III n'ont pas une signification absolue puisque leur détermination dépend d'un seuil de discrimination; tel qu'utilisé, un seuil élevé fournit des classements peu discriminants où beaucoup de bactéries ont le même rang; on a donc intérêt à choisir un seuil peu élevé; nous avons donc choisi pour les expérimentations un seuil de discrimination égale à .40, cette valeur paraissant fournir les meilleures identifications.

Sur chacune des 6 premières souches, nous avons réalisé plusieurs analyses de sensibilité sur les seuils de préférence, d'indifférence et de veto.

Les résultats obtenus sont les suivants :

si sp sv	souche 1	souche 2	souche 3	souche 4	souche 5	souche 6
10 15 90	C	E	C	C	C	C
10 15 60	C	E	C	E	C	C
0 0 90	C	E	E	E	C	C
0 0 60	E	E	E	E	X	X
50 60 90	C	E	C	-	-	-
60 70 90	C	E	E	C	C	C
70 80 90	E	E	E	E	C	C

(-) : signifie que l'analyse de sensibilité correspondante n'a pas été effectuée.

Constatations :

1. Le nombre élevé de résultats non corrects lorsque $si = sp = 0$ atteste de la nécessité de définir des seuils d'indifférence et de préférence > 0 ;
2. En fixant un seuil de veto élevé (≈ 90), on parvient, pour certaines souches, à obtenir des résultats corrects même si les seuils d'indifférence et de préférence sont très élevés.

Pour la souche 2, nous avons effectué des analyses de sensibilité supplémentaires en augmentant le seuil de veto à 99 avec $si = 10$ et $sp = 15$ ou $si = 5$ et $sp = 10$, ou en modifiant le seuil de discrimination à .30 ou à .70; toutes ces analyses de sensibilité ont fourni des

résultats erronés.

Pour les souches 7 à 15, en utilisant le triplet (10,15, 90), on a obtenu chaque fois l'identification correcte.

V.1.1.2 L'expérimentation vis-à-vis de la méthode Oreste.

La qualité de l'identification du germe inconnu par cette méthode va être déterminée en comparant le rang global de chaque bactérie;

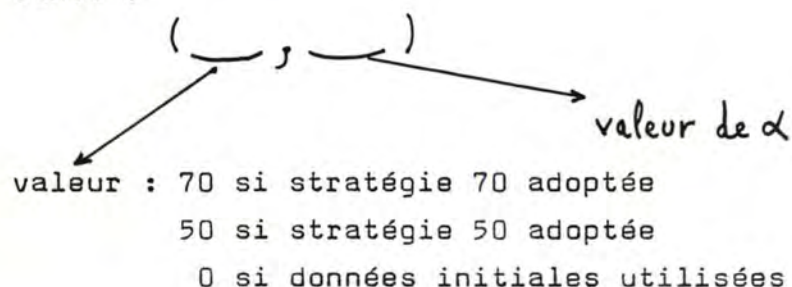
si r est le rang global de la bactérie *Klebsiella ozaenae*, on convient de dire que l'identification proposée par Oreste est :

- erronée (E) dès que r est strictement supérieur au rang global attribué à d'autres bactéries,
- suspecte (X) si plusieurs bactéries, dont *Klebsiella ozaenae*, ont le rang global minimum,
- correcte (C) dès que le rang r est inférieur à tous les autres rangs; nous indiquons entre parenthèses la valeur du seuil d'indifférence à partir de laquelle *Klebsiella ozaenae* et la bactérie dont le rang global lui est immédiatement supérieur sont considérées comme indifférentes par la procédure d'agrégation.

Pour chacune des 3 premières souches, on a effectué différentes analyses de sensibilité pour montrer l'influence sur le processus d'identification de :

1. la pondération ou non des évaluations initiales associées à la formule 1 par des formules de transformation;
2. la valeur attribuée au paramètre α associé à la distance city-block.

Les arguments entre parenthèses se trouvant dans la première colonne du tableau 1 ont la signification suivante :



	souche 1	souche 2	souche 3
(0,.25)	E	E	E
(0,.50)	E	E	E
(0,.80)	E	E	E
(50,.25)	C (0.00799)	X	C (.00769)
(50,.50)	C (0.00723)	X	C (.00894)
(50,.80)	C (0.0070)	X	C (.00921)
(70,.50)	E	X	E
(70,.80)	E	X	E

- tableau 1 -

Constatations :

1. α n'a apparemment pas beaucoup d'influence sur le processus d'identification. On a remarqué que lorsque α dépasse une certaine borne valant $\simeq 75$, les clas-

sements des bactéries en terme de rangs globaux sont les mêmes quelque soit α . Ceci peut s'expliquer par le fait qu'à ce moment, l'apport du poids de chaque critère dans le calcul de la distance devient négligeable. Dès lors, cette distance ne dépend plus principalement que de l'évaluation de chaque action; d'une valeur α à une autre, toutes les distances pour une même action et un même critère, sont proportionnelles.

2. La stratégie 50 fournit les meilleures identifications, on l'a dès lors choisie pour les souches 4 à 15.

L'identification de celles-ci, en choisissant une valeur α de .80, a fournit un diagnostic correct sauf pour la souche 7.

CONCLUSION.

Le but essentiel de ce travail était de mettre au point un logiciel qui permette, à partir d'une utilisation aisée, d'apporter à l'utilisateur des solutions pour l'aider à prendre une décision devant un problème multicritère donné. Ainsi, le logiciel a été conçu de telle sorte que les problèmes inhérents à la décision multicritère et aux méthodes multicritères soient résolus : au niveau de la saisie des préférences, où différents outils sont proposés tant pour faciliter le processus de formalisation des préférences chez le décideur, que pour les formaliser sous une forme acceptable par les méthodes multicritères; au niveau de la résolution d'un problème multicritère, où plusieurs méthodes ayant été proposées, il sera plus facile à l'utilisateur de prendre une décision sur base de résultats concordants issus des différentes méthodes.

Nous ne pouvons, néanmoins, prétendre avoir intégré tous les outils facilitant la saisie des préférences, ou toutes les méthodes multicritères; d'autres méthodes auraient pu, et seront vraisemblablement, intégrées à ce logiciel. Du point de vue des nouveaux outils qui pourront à l'avenir être mis au point et être supportés par le logiciel, on peut esquisser quelques directions vers lesquelles ils pourraient s'engager.

Le processus de désagrégation des préférences ne semble pas encore avoir reçu l'importance qu'on devrait lui accorder : peu de méthodes, jusqu'à présent, prétendant résoudre cet aspect du problème, d'importants développements sont à espérer de ce côté.

Les méthodes multicritères devraient être dirigées vers des méthodes plus souples tant au point de vue de la forme des préférences qu'elles acceptent - on peut songer à de nouvelles méthodes qui pourraient travailler, à l'instar de certaines variantes des méthodes UTA, sur des relations quelconques - que du point de vue des processus d'agrégation, où, à l'instar de la méthode Electre III, la relation de surclassement serait une relation floue, qui est donc moins rigide en terme du nombre de degrés de préférence qui lui sont associés.

En conclusion, nous constatons qu'il reste encore beaucoup de possibilités pour des travaux ultérieurs dans le domaine de l'aide à la décision multicritère. Néanmoins, nous espérons avoir contribué d'une manière ou d'une autre à une meilleure connaissance du sujet.

BIBLIOGRAPHIE.

- [1] E. Jacquet-Lagrèze, J. Siskos, "Une méthode de construction de fonctions d'utilité additives, explicatives d'une préférence globale", Cahier Lamsade, n° 16 (1978)
- [2] J. Siskos, "Les programmes UTA, manuel d'utilisation", Cahier Lamsade, n° 24 (1979)
- [3] E. Jacquet-Lagrèze, J. Siskos, "Assessing a set of additive utility functions for multicriteria decision-making, the UTA method", European Journal of Operational Research 10, 151-164 (1982)
- [4] J. Siskos, "Comment modéliser les préférences au moyen de fonctions d'utilité additives", R.A.I.R.O. Recherche Opérationnelle, vol. 14, n° 1, 53-82 (1980)
- [5] B. Roy, P. Vincke, J-P. Brans, "Aide à la décision multicritère", Revue Belge de Statistique, d'Informatique et de Recherche Opérationnelle, vol. 15, n° 4, décembre 1975
- [6] J. Fichet, J.P. Leclercq, Ph. Beyne et F.F. Rousset-Piette, "Aide à l'identification des entérobactéries par micro-ordinateur : un système basé sur des modèles d'aide à la décision multicritère", à paraître dans les Actes de la XII^{ème} Ecole d'Eté d'Informatique de l'AFCEt, Namur (Belgique), 12-24 juillet 1982

- [7] J. Fichefet, "Une méthode multicritère hiérarchique tenant compte d'une information incomplète sur l'importance des critères : la méthode HERMES", research paper 16/8, Institut d'Informatique, Namur.
- [8] E. Jacquet-Lagrèze, "Représentation de quasi-ordres et de relations probabilistes transitives sous forme standard et méthodes d'approximation", Math. Sci. Hum., 16^e année, n° 1963, 5-24 (1978)
- [9] Ph. Mataigne et Y. Perozzo, "Réalisation d'un logiciel graphique", Mémoire de Licence et Matrise en Informatique, Institut d'Informatique FNDP, 1982
- [10] Luce R.D., "Semi-orders and a theory of utility discrimination", Econometrica, vol. 24 (1956)
- [11] M. Roubens, "Analyse et agrégation des préférences : modélisation, ajustement et résumé de données relationnelles", Revue Belge de Statistique, d'Informatique et de Recherche Opérationnelle, vol. 20, n° 2, juin 1980
- [12] M. Roubens, "Médiane et méthodes multicritères ordinales", Revue Belge de Statistique, d'Informatique et de Recherche Opérationnelle, vol. 22, n° 2, juin 1982
- [13] M. Roubens, "Preference relations on actions and criteria in multicriteria decision-making", European Journal of Operational Research 10, 51-55 (1982)

- [14] J. Fichet, "la méthode hiérarchique des comparaisons par paires de L. Saaty",
Institut d'Informatique, Namur
- [15] Sidney Siegel, "Nonparametric statistics for the behavioral sciences",
Mc Graw-Hill Book Company, Inc., 202-213
- [16] R.L. Keeney, H. Raiffa, "Decisions with multiple objectives : preference and value tradeoffs",
John Wiley and Sons, 94-100 (1976)
- [17] P. Fishburn, "Methods of estimating additive utilities",
Management Science, vol. 13, n° 7, 435-453, March 1967
- [18] K.R. Oppenheimer, "A proxy approach to multiattribute decision making",
Management Science, vol. 24, n° 6, 675-689, (1978)
- [19] Fichet, J. "Eléments de programmation linéaire",
notes de cours, FNDP, 1981
- [20] Fichet, J. "Théorie des graphes et applications",
notes de cours, FNDP, 1980
- [21] P. Hansens, M. Anciaux, P. Vincke, "Quasi-kernels of outranking relations",
in H. Thiriez and S. Zionts, Eds. Multiple criteria decision making, lecture notes in Economics and Mathematical systems 30 (Springer, Berlin 1976)

- [22] P. Hansens, "Les procédures d'exploration de d'optimisation par séparation et évaluation",
29-65 in B. Roy (ed.). Combinatorial Prog. (Reidel, 1975)
- [23] J. Fichet, "Some insight into the nature of computer selection",
Revue Belge de Statistique, d'Informatique et de Recherche Opérationnelle, vol. 21, n° 3 (1981)
- [24] B. Roy, "Electre III : un algorithme de classements fondé sur une représentation floue des préférences en présence de critères multiples",
Cahiers du Centre d'Etudes de Recherche Operationnelle, vol. 20, 3-24 (1978)
- [25] J. Fichet, "Critique de la méthode MAL de J.J. Dujmović pour l'évaluation et la comparaison de systèmes complexes",
research papers 3/78, Institut d'Informatique, Namur
- [26] J.H.P. Paelinck, "Qualitative multiple criteria analysis, environmental protection and multiregional development",
Papers of regional science association, vol. 36, 59-74 (1976)
- [27] W.D. Cook, A.L. Saipé, "Comittee approach to priority planning : the median ranking method",
Cahiers du Centre d'Etudes de Recherche Opérationnelle, 18, 337-351 (1976)

- [28] R.D. Armstrong et W.D. Cook, "A comittee ranking problem using ordinal utilities",
Revue Belge de Statistique, d'Informatique et de Recherche Opérationnelle, vol. 21, n° 4, 3-23 (1981)
- [29] J.F. Marcotorchino et P. Michaud, "Optimisation en analyse ordinale des données",
Masson, Paris (1976)
- [30] J. Larmouth, "Fortran 77 portability",
Software-practice and experience, vol. 11, 1071-1117 (1981)

FACULTES UNIVERSITAIRES NOTRE-DAME DE LA PAIX (Namur).

MISE AU POINT D'UN SYSTEME
CONVERSATIONNEL POUR L'AIDE
A LA DECISION MULTICRITERE.

ANNEXES.

Luc MARY.

Année académique 1982 - 1983.

Annexes :Annexe I : Eléments de la théorie de l'utilité.

I.1 Notions associées à une relation de préférence. A1

I.2 Fonctions d'utilité. A5

Annexe II : les autres méthodes multicritères implémentées.

II.1 La méthode Electre III. A8

II.1.1 Objectif. A8

II.1.2 Principe de construction de la relation de surclassement floue. A10

II.1.2.1 Calcul du degré de crédibilité d_j spécifique au critère j . A10

II.1.2.2 L'indice de concordance. A12

II.1.2.3 Le seuil de veto. A12

II.1.2.4 L'indice de discordance. A12

II.1.2.5 Le degré de crédibilité d . A13

II.1.3 L'obtention des classements des actions. A15

pages :

II.1.3.1 Le concept de λ -qualification d'une action.	A15
II.1.3.2 Le principe d'obtention des 2 pré- ordres.	A16
II.2 La méthode des comparaisons par paires de Saaty.	A20
II.2.1 Rappel sur les ensembles et la notion de hiérarchie.	A20
II.2.2 Problématique.	A23
II.2.3 Les étapes de la méthode.	A25
II.3 La méthode "MAL" de Dujmović.	A26
II.3.1 Présentation de la méthode.	A26
II.3.2 Le principe de l'agrégation.	A27
II.4 La méthode des rangements médians.	A32
II.4.1 Objectifs.	A32
II.4.2 Intérêt de la méthode.	A32
II.4.3 Définition de distance et adhésion.	A34
II.4.4 Méthodes pour rechercher les préordres médians.	A36

pages :

II.4.5 Définitions et théorèmes préliminaires.	A37
II.4.6 Principe de détermination du préordre médian.	A41
II.4.7 Algorithme.	A43

Annexe III : Description des fichiers utilisés.

III.1 L'enregistrement des rapports de préférence.	A 49
III.1.1 L'enregistrement des relations de préférence.	A. 49
III.1.1.1 Les fichiers associés aux rapports de préférence partielle entre les actions.	A50
III.1.1.2 Les fichiers associés aux rapports de préférence entre les critères.	A54
III.1.1.3 Les fichiers associés aux rapports de préférence globale entre les actions et aux relations de sur-classement déterministes.	A57
III.1.2 Les fichiers associés à la méthode de de rangement par des comparaisons de Saaty.	A60
III.1.2.1 Les fichiers associés aux rapports de préférence partielle entre les actions.	A61

pages :

III.1.2.2 Les fichiers associés aux rapports de préférence entre les critères.	A6 4
III.1.2.3 Les fichiers associés aux rapports de préférence globale entre les actions.	A6 4
III.1.3 Les fichiers associés à la construction d'une fonction d'utilité pour déterminer les rapports de préférence partielle entre les actions.	A6 6
III.2 L'enregistrement des paramètres et résultats associés à chaque méthode multicritère.	A6 8
III.2.1 Les fichiers associés à la méthode Oreste.	A7 0
III.2.2 Les fichiers associés à la méthode Electre I.	A7 2
III.2.3 Les fichiers utilisés par les méthodes Oreste et Electre I.	A7 3
III.2.4 Les fichiers associés à la méthode Electre II.	A7 5
III.2.5 Les fichiers associés à la méthode Electre III.	A7 7

pages :

III.2.6 Les fichiers associés à la méthode hiérarchique des comparaisons par paires de Saaty.	A 79
III.2.7 Les fichiers associés à la méthode des rangements médians.	A 80
III.2.8 Les fichiers associés à la méthode UTA I.	A 83
III.2.9 Les fichiers associés à la méthode UTA II.	A 84
III.2.10 Les fichiers communs aux méthodes UTA I et UTA II.	A 85
III.2.11 Les fichiers associés à la méthode MAL.	A 90

Annexe 1 : Éléments de la théorie de l'utilité.

I.1 Notions associées à une relation de préférence.

1. Soit un ensemble fini A à n éléments et une relation B de préférence sur A .

On convient d'écrire aBb si l'on peut énoncer la proposition "a est préféré à b". Une telle proposition n'exclut pas que b puisse être préféré à a .

La partie stricte de B est notée P (préférence) : relation supposée asymétrique ($\forall a, b : aPb$ si et seulement si aBb et non (bBa)).

La partie d'indifférence de B est notée I : relation supposée symétrique ($\forall a, b : aIb$ si et seulement si aBb et bBa).

La partie d'incomparabilité de B est notée R : relation supposée irréflexive et symétrique ($\forall a, b : aRb$ si et seulement si non (aBb) et non (bBa)).

Par convention, on note $B = (P, I, R)$ ou $B = PUIR$; ces 3 relations sont supposées mutuellement exclusives et telles que :

$\forall a, b \in A : aPb \text{ ou } bPa \text{ ou } aIb \text{ ou } aRb$ (ou = ou exclusif).

2. Si $R = \emptyset$, la relation B est un quasi-ordre lorsque :

(a) $\forall a, b, c, d \in A : aPb, bIc, cPd \Rightarrow aPd$;

(b) $\forall a, b, c, d \in A$: les deux situations (aPb, bPc) et (aId, dIc) sont incompatibles.

Dans un quasi-ordre, la préférence P est donc transitive (cfr. (a) avec $b = c$) mais l'indifférence n'est pas nécessairement transitive.

3. Si $R = \emptyset$, la relation B est un préordre total lorsque :

- (a) $\forall a, b, c \in A : aBb \text{ et } bBc \Rightarrow aBc$;
- (b) $\forall a, b, c, d \in A$: les deux situations (aPb, bPc) et (aId, dId) sont incompatibles.

Dans un préordre, la préférence P et l'indifférence I sont donc transitives.

4. Si $R = \emptyset$ et $I = \emptyset$, la relation B est un ordre total lorsque :

$$\forall a, b, c \in A : aBb \text{ et } bBc \Rightarrow aBc$$

Si la relation B n'est pas un quasi-ordre, on convient de dire qu'elle est quelconque.

5. Représentation de la relation B :

(a) l'information associée à B peut être stockée à l'aide d'un codage binaire (0 ou 1) dans une matrice d'adjacence M définie comme suit :

$$\begin{aligned} \forall a, b \in A : M(a, b) &= 1 \text{ si } aBb, \\ M(a, b) &= 0 \text{ si non } (aBb). \end{aligned}$$

On constate que :

aPb si et seulement si $M(a, b) = 1$ et $M(b, a) = 0$
 aIb si et seulement si $M(a, b) = 1$ et $M(b, a) = 1$
 aRb si et seulement si $M(a, b) = 0$ et $M(b, a) = 0$

- (b) si la relation B sur A est un quasi-ordre, la classe d'équivalence d'un élément quelconque $a \in A$ est l'ensemble A_i des éléments auxquels a est lié par I, c'est-à-dire :

$$A_i = \left\{ b \in A \mid b I a \right\}; \text{ le nombre d'éléments}$$

$\in A_i$, noté n_i , est appelé le cardinal de A_i .

Si on décompose le quasi-ordre B en m classes d'équivalence ordonnées totalement par ordre décroissant de préférence A_1, \dots, A_m :

- lorsque B est un préordre total, on définit le rang moyen r de $x \in A_i$ comme suit :

$$r(x) = \sum_{j=1}^{i-1} n_j + \frac{n_i + 1}{2}.$$

Constatant que pour 2 éléments $x \in A_i$ et $y \in A_j$ (avec $j > i$), $r(x) < r(y)$, si l'on veut que $r(x)$ soit supérieur à $r(y)$, on définit alors le rang moyen de x comme suit :

$$r(x) = n - \sum_{j=1}^{i-1} n_j + \frac{n_i + 1}{2};$$

- lorsque B est un quasi-ordre, on peut définir une notion similaire à celle de rang.

Remarquant que la seule différence entre un quasi-ordre et un préordre total réside dans l'existence, pour un quasi-ordre, d'éléments appartenant simultanément à deux classes d'équivalence adjacentes,

nous avons, dès lors, convenu de définir,

$\forall x \in B$, sa position $p(x)$ tel que :

$p(x)$ = rang (moyen ou non) de x si x appartient à une seule classe d'équivalence

ou

$p(x) = -$ (rang des éléments appartenant à A_i)
si $x \in$ à 2 classes d'équivalence A_i et A_{i+1} .

6. Distance et tau τ de Kendall.

Si $R = \{R_{ij}\}$ et $R^* = \{R^*_{ij}\}$ sont 2 matrices d'adjacences associées aux relations B et B^* , on définit le tau τ de Kendall comme suit :

$$\tau = 1 - \frac{4d_K(R, R^*)}{n(n-1)}$$

où $d_K(R, R^*)$ est la distance de Kendall définie comme suit :

$$d_K(R, R^*) = \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n |R_{ij} - R^*_{ij}|$$

Le tau de Kendall varie de -1 à +1; plus τ est proche de 1, et plus les deux relations B et B^* sont "proches" l'une de l'autre.

I.2 Fonctions d'utilité.

La définition retenue de fonction d'utilité est celle proposée par Von Neumann et Morgenstern [25] ; cette définition prend appui sur les axiomes suivants :

1. les axiomes de départ sont :

- un ensemble A d'éléments
- une relation de préférence sur A
- une opération $(\alpha, x; 1 - \alpha, y)$ telle que $(\alpha, x; 1 - \alpha, y) \in A$ si, et seulement si, $\alpha \in (0,1)$, c'est-à-dire $0 < \alpha < 1$, et $x, y \in A$.

Exemple :

si $A = \mathbb{R}^1$, l'opération $(\alpha, x; 1 - \alpha, y)$ pourrait être celle qui, à $\alpha \in (0,1)$ et à $x, y \in \mathbb{R}^1$, fait correspondre le nombre $\alpha x + (1 - \alpha)y$.

2. les axiomes de Von Neumann et Morgenstern.

Dans ce qui suit, nous écrirons, $\forall x, y \in A$:

$$x \leq y$$

pour signifier que x'' n'est pas préféré à y'' .

Nous poserons :

$$x < y \text{ (} y \text{ est préféré à } x \text{)} \iff y(\text{non } \leq)x,$$

$$x \sim y \text{ (} x \text{ est indifférent à } y \text{)} \iff x \leq y \text{ et } y \leq x,$$

$$y \geq x \text{ (} y \text{ est au moins aussi désiré que } x \text{)} \iff x \leq y,$$

$$y > x \iff x < y$$

Axiomes.

1. $\forall x, y \in A : x < y \text{ ou } y > x \text{ ou } x \sim y.$
2. Si $x < y$ et $y < z$, alors : $x < z$.
3. Si $x < y$, alors $x < (\alpha, x; 1 - \alpha, y)$ pour tout $\alpha \in (0, 1)$.
4. Si $x > y$, alors $x > (\alpha, x; 1 - \alpha, y)$ pour tout $\alpha \in (0, 1)$.
5. Si $x < y < z$, alors $\exists \alpha \in (0, 1)$ tel que $(\alpha, x; 1 - \alpha, z) < y$.
6. Si $x > y > z$, alors il existe $\alpha \in (0, 1)$ tel que $(\alpha, x; 1 - \alpha, y) > y$.
7. $(\alpha, x; 1 - \alpha, y) \sim (1 - \alpha, y; \alpha, x)$ pour tout $\alpha \in (0, 1)$.
8. $(\alpha, (\beta, x; 1 - \beta, y); 1 - \alpha, y) \sim (\alpha\beta, x; 1 - \alpha\beta, y)$ pour tous $\alpha, \beta \in (0, 1)$.
9. Si $x \sim y$, alors $(\alpha, x; 1 - \alpha, z) \sim (\alpha, y; 1 - \alpha, z)$ pour tout $\alpha \in (0, 1)$.

Ces axiomes permettent de garantir l'existence d'une fonction numérique $U : X \rightarrow \mathbb{R} : x \rightarrow U(x)$ appelée fonction d'utilité - au sens de Von Neumann et Morgenstern - et telle que :

$$\begin{aligned}
 x < y &\iff U(x) < U(y), \\
 U(\alpha, x; 1 - \alpha, y) &= \alpha U(x) + (1 - \alpha) U(y), \\
 \forall \alpha &\in (0, 1).
 \end{aligned}$$

De plus, si U et V sont des fonctions d'utilité sur A , alors il existe $a, b \in \mathbb{R}$, avec $a > 0$, tels que :

$$V(x) = a U(x) + b,$$

ce qui signifie que la fonction d'utilité est unique, à une transformation linéaire positive près.

Supposons que la relation de préférence \leq soit définie sur

$A = \prod_{i=1}^n A_i$, avec $A_i \neq \emptyset$, $\forall i$. On dit qu'une fonction

d'utilité $U(x) = U(x_1, x_2, \dots, x_n)$ associée à \leq est additive si elle peut être écrite sous la forme :

$$U(x_1, x_2, \dots, x_n) = U_1(x_1) + U_2(x_2) + \dots, \\ + U_n(x_n)$$

où U_i est une fonction numérique définie sur A_i .

Annexe II : les autres méthodes multicritères implémentées.

II.1 La méthode Electre III [24] .

II.1.1 Objectif.

Les critiques suivantes s'imposent aux méthodes d'agrégation évoquées jusqu'à présent :

- 1) si $u (\geq 0)$ désigne la différence $X_i(a) - X_i(b)$ entre les évaluations des actions a et b pour le critère i , ces méthodes supposent que :
 - a est préféré à b pour ce critère dès que u est différent de 0,
 - a est indifférent à b pour ce critère quand $X_i(a) = X_i(b)$.

Or, il se peut que :

- u ne soit pas significativement différent que pour décider d'une préférence de a sur b pour le critère i ; il existe donc une limite $q (\geq 0)$, appelée seuil d'indifférence, qui, tant que $u \leq q$, permet de dire que a et b sont indifférents pour le critère i .
- u ne soit pas suffisamment important que pour décider de la préférence de a sur b ; il existe donc une limite inférieure $s \geq q$, appelée seuil de préférence, qui si $u \geq s$, permet de dire que a est préféré à b pour le critère i .
- $q < u \leq s$; il y a alors hésitation entre l'indifférence et la préférence; on dit alors que a est

faiblement préféré à b.

L'introduction de ces 2 seuils permet de définir 3 rapports de préférence entre 2 actions pour un critère i donné :

- l'indifférence entre a et b si $0 \leq u \leq q$,
- la préférence stricte de a sur b si $s < u$,
- la préférence faible de a sur b si $q < u \leq s$ (formule 1).

Remarques.

1. Les 2 seuils s et q dépendent du critère i envisagé et éventuellement aussi de la valeur $X_i(a)$ par rapport à laquelle ils sont repérés; c'est pourquoi, on les notera si $[X_i(a)]$ et qi $[X_i(a)]$.

2. Toute fonction seuil $s(\lambda)$ définie sur $[x, y]$ doit vérifier la condition : $\frac{s(\lambda) - s(\lambda')}{\lambda - \lambda'} \geq -1$

$$\forall \lambda, \lambda' \in [x, y].$$

- 2) La relation de surclassement déterministe obtenue par ces méthodes est une relation :
 - pauvre : si une action a est surclassée par une action b, on ne sait pas suivant quelle intensité (faible, forte, moyenne,...)elle l'est.
 - arbitraire : en ce sens que parfois, le surclassement d'une action par une autre est si faible qu'en modifiant légèrement certains paramètres associés au processus d'agrégation, on aboutirait à un non-surclassement.

Ces inconvénients suggèrent de remplacer la recherche d'une relation de surclassement déterministe par celle d'une relation de surclassement floue où l'on s'attache

à déterminer suivant quel degré de crédibilité une action en surclasse une autre.

Electre III répond à ces critiques en s'appuyant sur les concepts de seuils de préférence, de seuils d'indifférence et de relation de surclassement floue. Elle aboutit, comme Electre II, à classer les actions suivant des préordres.

La méthode Electre III procède ainsi en 3 étapes :

- construction d'une relation de surclassement floue,
- obtention de deux préordres à partir de la relation de surclassement,
- comparaison des deux préordres obtenus et élaboration d'un rangement final suivant le même principe que pour Electre II.

II.1.2 Principe de construction de la relation de surclassement floue.

II.1.2.1 Calcul du degré de crédibilité d_j spécifique au critère j .

Pour comparer deux actions a et b pour un critère donné, plutôt que d'utiliser leurs évaluations $X_j(a)$ et $X_j(b)$ comme telles, la procédure d'agrégation utilise un degré de crédibilité propre au critère j : d_j .

En fonction de la formule 1 du point II.1.1 et de la propriété 3 associée au degré de crédibilité énoncée au point III.1, d_j est défini comme suit :

$$d_j(b,a) = 1 \text{ si } X_j(b) - X_j(a) \geq 0$$

$$d_j(a,b) = 0 \text{ si } X_j(b) - X_j(a) > s_j [X_j(a)]$$

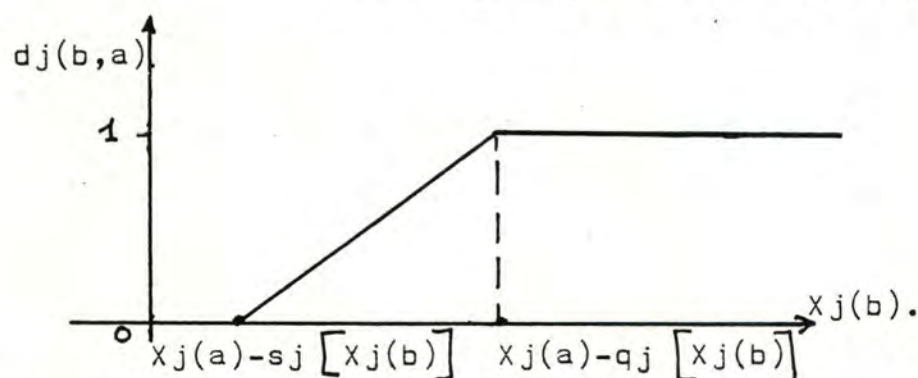
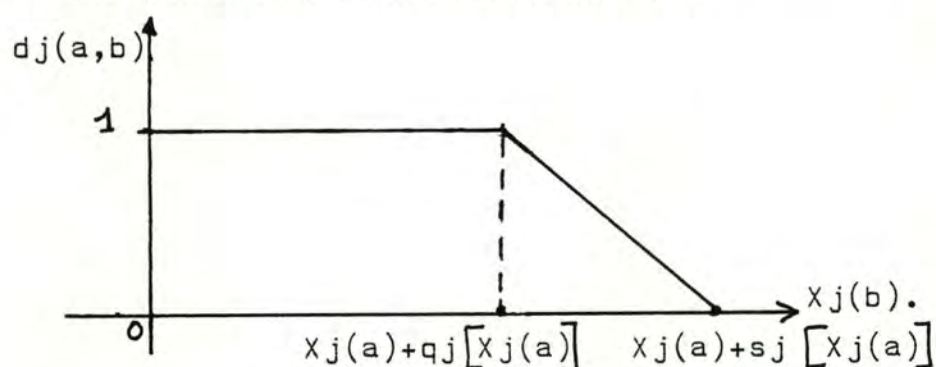
$$d_j(a,b) = 1 \text{ si } 0 \leq X_j(b) - X_j(a) \leq q_j [X_j(a)] ;$$

Lorsque $q_j [X_j(a)] \neq s_j [X_j(a)]$ et

$$q_j [X_j(a)] < X_j(b) - X_j(a) < s_j [X_j(a)] ,$$

on fixe $d_j(a,b)$ par extrapolation linéaire.

$d_j(a,b)$ et $d_j(b,a)$ peuvent alors être représentés graphiquement en fonction de $X_j(b)$:



L'utilisation du degré de crédibilité d_j pour comparer deux actions a et b sur le critère j nécessite la définition de seuils et d'indices distincts de ceux utilisés pour Electre I ou Electre II.

II.1.2.2 L'indice de concordance : chacun des critères j contribue à la concordance du surclassement de a sur b en fonction de son poids pondéré par le degré de crédibilité pour j du surclassement de a sur b : $d_j(a,b)$.

L'indice de concordance est alors :

$$C(a,b) = \sum_{j=1}^{k1} p_j \times d_j(a,b).$$

II.1.2.3 Le seuil de veto : j est en discordance avec le surclassement de a sur b si j n'est pas en concordance

avec ce surclassement, c'est-à-dire si

$d_j(a,b) = 0 \iff X_j(b) > X_j(a) + s_j [X_j(a)]$; plus $X_j(b) - X_j(a) - s_j [X_j(a)]$ est grand, et plus la discordance est forte.

Le seuil de veto ou seuil de discordance $v_j [X_j(a)]$ est alors défini comme la valeur de la différence $X_j(b) - X_j(a)$ à partir de laquelle il apparaît prudent de refuser toute crédibilité au surclassement de a sur b ; donc si $X_j(b) > X_j(a) + v_j [X_j(a)]$, alors $d(a,b) = 0$.

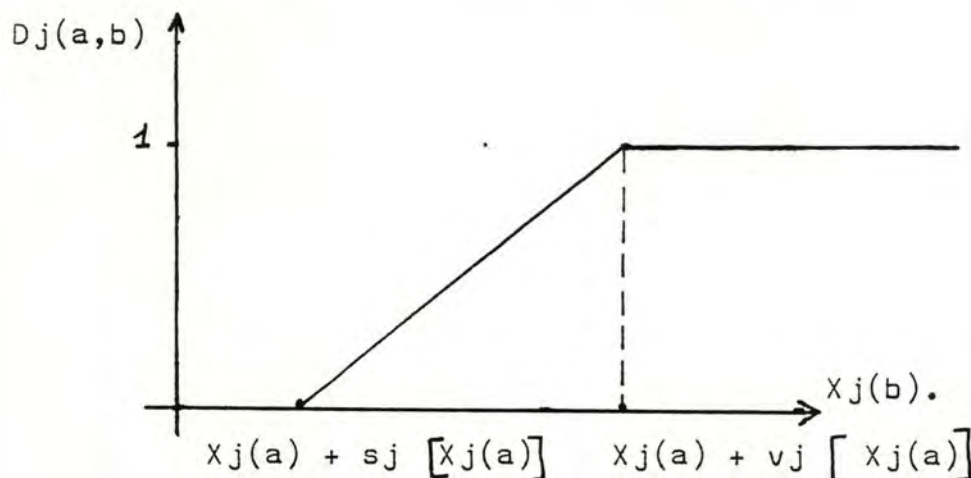
Ce seuil est fonction du critère j , et éventuellement de $X_j(a)$; il doit être choisi \geq à $s_j [X_j(a)]$.

II.1.2.4 L'indice de discordance : il mesure l'intensité de la discordance du critère j vis-à-vis de l'opinion que a surclasse b .

$D_j(a,b)$ est une fonction qui :

- doit être monotone non décroissante de $X_j(b)$
- vérifie $D_j(a,b) \cdot d(a,b) = 0$
- plafonne à 1 quand $X_j(b) > X_j(a) + v_j [X_j(a)]$.

La valeur de D_j dans l'intervalle $]x_j(a) + s_j [x_j(a), x_j(a) + v_j [x_j(a)[$ est fixée par interpolation linéaire (cfr. figure 1); l'indice vaut par conséquent :

$$D_j(a,b) = \min \left[1, \max \left(0, \frac{x_j(b) - x_j(a) - s_j [x_j(a)]}{v_j [x_j(a)] - s_j [x_j(a)]} \right) \right]$$


- figure 1 -

II.1.2.5 le degré de crédibilité d.

Le calcul du degré de crédibilité d associé à la relation de surclassement floue dépend de deux éléments :

- l'indice de concordance $C(a,b)$
- les indices de discordance $D_j(a,b)$, et conséquemment, le degré de crédibilité $d_j(a,b)$.

3 cas sont à distinguer pour déterminer d :

1er cas : tous les indices de discordance sont nuls; il semble alors logique de choisir $C(a,b)$ comme degré de crédibilité d .

2ème cas : il existe des $D_j \neq 0$, peu élevés vis-à-vis de $C(a,b)$; par conséquent, les $d_j(a,b)$ corres-

pondants sont nuls, et l'indice de concordance s'en trouve affecté légèrement; il reste un reflet correct de d .

3ème cas : il existe des $D_j \neq 0$ tel que $D_j(a,b) > C(a,b)$.
 Supposons que seul le critère j soit dans ce cas, et examinons comment doit varier d en fonction de $D_j(a,b)$:

- si $D_j(a,b) = C(a,b)$: on a $d(a,b) = C(a,b) < 1$
- si $D_j(a,b) = 1$, $d(a,b) = 0$.

De plus, $d(a,b)$ est une fonction non-décroissante de $X_j(b)$, donc aussi de $D_j(a,b)$.
 Si on admet la décroissance linéaire entre $C(a,b)$ et 1, on a :

$$d(a,b) = C(a,b) \times \frac{1 - D_j(a,b)}{1 - C(a,b)} .$$

S'il existe plusieurs critères dans cette situation, il est logique d'admettre que leurs effets se conjuguent, d'où la formule suivante pour déterminer $d(a,b)$.

Si $F(a,b) = \left\{ i \in K \text{ tel que } D_i(a,b) > C(a,b) \right\}$:

- si $F(a,b) = \emptyset$, $d(a,b) = C(a,b)$
- sinon, $d(a,b) = C(a,b) \times \prod_{i \in F} \frac{1 - D_i(a,b)}{1 - C(a,b)} .$

Remarque : compte tenu des interpolations linéaires associées au calcul de D_j , d_j et d , on ne peut accorder une signification trop absolue aux degrés obtenus; seul l'ordre de grandeur est porteur d'une signification.

Si $D(a,a') > D(b,b')$, on ne pourra, par conséquent, conclure que le surclassement de a

sur a' est strictement plus crédible que le surclassement de b sur b' ; on est alors obligé d'introduire un seuil de préférence $s(\lambda)$, appelé seuil de discrimination, associé au critère "degré de crédibilité" et défini sur $\lambda \in [0,1]$.

Ainsi, si $d(a,a') > d(b,b') + s[d(a,a')]$, alors le surclassement de a' par a est strictement plus crédible que le surclassement de b sur b' ;

sinon, la crédibilité des 2 surclassements est du même ordre de grandeur.

II.1.3 L'obtention des classements des actions.

II.1.3.1 Le concept de λ -qualification d'une action.

Pour un préordre complet P , on peut définir pour toute action a de P un nombre, appelé qualification de a , qui égale la différence entre le nombre d'actions auxquelles a est préféré, appelé puissance de a , et le nombre d'actions qui lui sont strictement préférées, appelé faiblesse de a ; cette qualification de a est constante pour une classe d'équivalence de P , et est représentative de la position qu'occupe a dans le préordre.

Si on veut introduire une notion similaire pour une relation de surclassement floue, on doit utiliser le seuil de discrimination $s(\lambda)$. Il permettra de déterminer les actions qui sont préférées à a , et celles auxquelles a est strictement préféré. Ainsi, si B est un sous-ensemble de M , sont définies :

- la λ -puissance de a dans B :

$$p_B^\lambda(a) = \left| \left\{ b \in B, d(a,b) > \lambda, d(a,b) > d(b,a) + s[d(a,b)] \right\} \right|$$

- la λ -faiblesse de a dans B :

$$f_B^\lambda(a) = |\{b \in B, d(b,a) > \lambda, d(b,a) > d(a,b) + s[d(b,a)]\}|$$

- la λ -qualification de a dans B :

$$q_B^\lambda(a) = p_B^\lambda(a) - f_B^\lambda(a)$$

Il est clair que plus $q_B^\lambda(a)$ est élevé, et plus a est bien classé.

II.1.3.2 Le principe d'obtention des 2 préordres.

Le concept de λ -qualification va être utilisé pour obtenir les 2 préordres à partir de la relation de surclassement floue. La valeur λ n'est pas fixée une fois pour toutes, mais diminue au fur et à mesure de l'état d'avancement de la procédure grâce au concept de niveau de séparation (cfr. remarque 1).

Si λ_0 est la valeur maximale qu'atteint le degré de crédibilité sur l'ensemble M , on fixe le premier palier

$\lambda_1 = \lambda_0 - s(\lambda_0)$; si D_1 est l'ensemble des actions qui ont une λ_1 -qualification maximale, ces actions sont candidates à la première classe $\overline{C1}$ du préordre \overline{P} .

Pour réduire leur nombre, on fixe un second palier

$\lambda_2 < \lambda_1$; on obtient $D_2 \subset D_1$ en recherchant la λ_2 -qualification maximale sur les actions de D_1 .

Le processus itératif consistant à rechercher un sous-ensemble d'actions de plus en plus réduit, ayant une qualification maximum pour des paliers de plus en plus bas, est appelé distillation descendante; en continuant

le processus, on obtient un distillat final D_k lorsque $|D_k| = 1$ ou $\lambda_k = 0$.

D_k constitue la première classe du préordre \bar{P} ; pour obtenir la classe \bar{C}_2 , on applique le même processus sur le sous-ensemble des actions non-encore classées $B = A \setminus \bar{C}_1$; on définit ainsi une chaîne de distillation descendante qui s'interrompt lorsque la dernière classe est déterminée.

Pour obtenir un second préordre \underline{P} , au lieu de rechercher sans cesse la première classe du sous-ensemble B des actions non encore classées en s'intéressant aux λ -qualifications maximum, il est tout aussi naturel de progresser en recherchant la dernière classe des actions non encore classées, en prenant appui sur les actions de λ -qualification minimum.

On définit ainsi une chaîne de distillations ascendantes où l'on obtient en premier lieu la dernière classe de \underline{P} , et en dernier lieu la première classe de \underline{P} .

Remarques sur les étapes de l'algorithme.

1. Le niveau de séparation dans B à partir de λ est une valeur définie par :

$$d_B^\lambda = \begin{cases} \text{MAX } d(a,b) & \text{si } V = \{(a,b) \mid a,b \in B, d(a,b) < \lambda\} \neq \emptyset \\ a,b \in V \\ 0 & \text{si } V = \emptyset \end{cases}$$

2. Obtention du premier distillat de $B \subset A$.

Si λ_0 est la valeur maximum qu'atteint le degré de crédibilité sur le sous-ensemble B des actions, on calcule :

- $\lambda_1 = d_B \lambda_0[B] - s[\lambda_0(B)]$
- $q_B(a) \forall a \in B$
- - pour une distillation ascendante, $q = \min_{a \in B} q_B^{\lambda_1}(a)$
- - pour une distillation descendante, $q = \max_{a \in B} q_B^{\lambda_1}(a)$
- $D1 = \{a \mid a \in B, q_B^{\lambda_1}(a) = q\}$.

$D1$ est le premier distillat, associé à λ_1 , obtenu.

3. Passage du k ème au $k+1$ ème distillat.

Si λ_k est associé au distillat D_k , on calcule :

- $\lambda_{k+1} = d_{D_k} \lambda_k - s(\lambda_k)$
- $q_{D_k}^{\lambda_{k+1}}(a) \forall a \in D_k$
- - pour une distillation ascendante, $q = \min_{a \in D_k} q_{D_k}^{\lambda_{k+1}}(a)$
- - pour une distillation descendante, $q = \max_{a \in D_k} q_{D_k}^{\lambda_{k+1}}(a)$
- $D_{k+1} = \{a \mid a \in D_k, q_{D_k}^{\lambda_{k+1}}(a) = q\}$

4. Les paramètres pouvant faire l'objet d'une analyse de sensibilité sont :

- les seuils d'indifférence
- les seuils de préférence
- les seuils de veto
- la fonction seuil de discrimination.

II.2 La méthode hiérarchique des comparaisons par paires de Saaty [14] .

Nous avons vu (cfr. point II.1.2) comment il était possible, à partir de comparaisons par paires, de déterminer les poids des critères et les évaluations des actions; le concept de hiérarchie permet d'utiliser les poids et les évaluations ainsi obtenus pour établir un classement global des actions.

II.2.1 Rappel sur les ensembles et la notion de hiérarchie.

Définition 1 : un ensemble ordonné est un couple (A, \leq) où A est un ensemble et où \leq est une relation d'ordre dans A .

Notation : nous dirons que :

- $x < y$ lorsque $x \leq y$ et $x \neq y$
- y couvre x lorsque $x < y$ et qu'il n'existe aucun $z \in A$ tel que $x < z < y$.

Un ensemble ordonné sera représenté par un graphe (A, U) où U est l'ensemble des arcs, défini par :

(x, y) est un arc d'extrémités initiale x et finale y
 $y \Leftrightarrow y < x$ c'est-à-dire



Définition 2 : un ensemble ordonné (A, \leq) est dit totalement ordonné ou est appelé une chaîne lorsque \leq est totale.
Un ensemble non totalement ordonné est dit partiellement ordonné.

Définition 3 : un sous-ensemble B d'un exemple ordonné (A, \leq) est dit borné supérieurement si :
 $\exists y \in B : x \leq y, \forall x \in B.$
y est alors appelé borne supérieure de B

Définition 4 : un sous-ensemble B d'un ensemble ordonné (A, \leq) est dit avoir un supremum lorsqu'il est borné supérieurement et lorsque l'ensemble S de ses bornes supérieures possède un élément s tel que
 $\forall t \in S, t \leq s.$
un tel élément s est unique et appelé supremum de B : $s = \sup B$

Notation : si (A, \leq) est un ensemble ordonné, nous poserons, $\forall x \in A$:

$$\begin{aligned} x^- &= \left\{ y \in A \mid x \text{ couvre } y \right\} \\ x^+ &= \left\{ y \in A \mid y \text{ couvre } x \right\} \end{aligned}$$

Définition 5 : soit H un ensemble fini partiellement ordonné dont le supremum est l'élément $b \in H$. On dit que (H, \leq) est une hiérarchie s'il satisfait aux conditions suivantes :

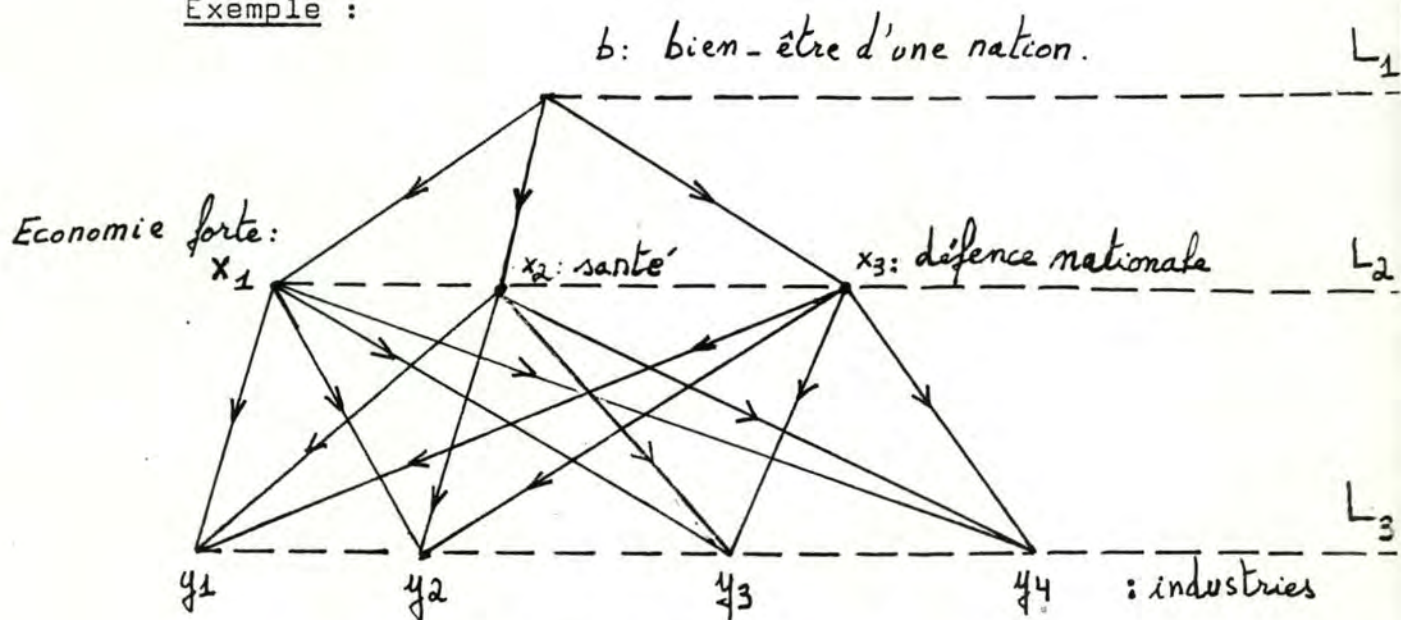
- a) il existe une partition de H en ensembles $L_k, k = 1, \dots, h$
où $L_1 = \{b\}$;
b) $x \in L_k \Rightarrow x^- \subset L_{k+1}, k = 1, \dots, h-1$;
c) $x \in L_k \Rightarrow x^+ \subset L_{k-1}, k = 2, \dots, h.$

Les ensembles L_k sont appelés les niveaux de la hiérarchie.

De plus, une hiérarchie est complète si :

$$\forall x \in L_k : x^+ = L_{k-1}, k = 2, \dots, h.$$

Exemple :



- figure 1 -

Le graphe de la figure 1 est une hiérarchie complète, puisque $x_1^+ = x_2^+ = x_3^+ = \{b\} = L_1$.

$$y_1^+ = y_2^+ = y_3^+ = y_4^+ = \{x_1, x_2, x_3\} = L_2.$$

Si on supprime l'arc (x_1, y_1) , la hiérarchie n'est plus complète.

Définition 6 : soient H une hiérarchie, et $x \in H$. Une fonction de priorité des éléments couverts par x est une fonction $W_x : x^- \rightarrow [0, 1]$ telle que $\sum_{y \in x^-} W_x(y) = 1$.

II.2.2 Problématique.

On constate qu'une hiérarchie peut servir à décrire la réalisation d'un objectif à partir de sous-objectifs. Dans le cas de la figure 1, l'objectif principal est le bien-être d'une nation; le 2e niveau contient trois objectifs : économie forte, santé, défense nationale; le 3e niveau correspond aux industries : ce sont les sous-objectifs.

On peut vouloir connaître l'impact de chaque industrie sur le bien-être de la nation par l'intermédiaire des objectifs du deuxième niveau.

Dans le cas d'un problème multicritère d'agrégation , l'objectif principal peut être le classement global des actions; le 2e niveau contient les k_1 critères intervenant dans la détermination du classement, et le 3e niveau contient les actions.

Connaissant - pour chaque critère y_j du second niveau, un indice de son importance relative $\overline{W}_x(y_j)$
 - pour chaque action z_i , une évaluation traduisant son importance relative dans la réalisation d'un critère donné y_j : $W_{y_j}(z_i)$,
 le problème est de déterminer un indice d'importance w_j de chaque action qui reflète son importance, et donc sa position, dans la réalisation du classement global.

La problématique dans le cas général est la suivante :
 soit H une hiérarchie de niveaux L_1, \dots, L_h et supposons que soit définie, pour tout $x \in H$, une fonction de priorité W_x des éléments de x^- . Soient $x \in L_\alpha$ et $S \subset L_\beta$
 ($\alpha < \beta$).

Comment définir une fonction de priorité :

$w_{x,S} : S \rightarrow [0,1]$ reflétant les priorités w_y des niveaux L_k , $k = \alpha, \dots, B-1$?

ou encore plus spécifiquement,

étant donné un système dont l'objectif principal est b et dont l'ensemble des activités de base est L_k , quelles sont les priorités des activités de L_k dans la réalisation de l'objectif b ?

Saaty propose de procéder comme suit pour les hiérarchies complètes.

Soient $x \in L_{k-1}$, $L_k = \{y_1, \dots, y_{m_k}\}$,
 $L_{k+1} = \{z_1, \dots, z_{m_{k+1}}\}$,

alors, la fonction de priorité des éléments de L_{k+1} par rapport à x est la fonction $W : L_{k+1} \rightarrow [0,1]$

définie par : $w(z_i) = \sum_{j=1}^{m_k} w_x(y_j) \cdot w_{y_j}(z_i)$ $i = 1, \dots, m_{k+1}$

ou, sans forme matricielle :

$$w_i = \sum_{j=1}^{m_k} b_{ij} \bar{w}_j \quad i = 1, \dots, m_{k+1}$$

ou

$$W = B\bar{W}$$

Le vecteur W et la matrice B ainsi définis sont appelés respectivement vecteur de priorité et matrice de priorité du niveau L_{k+1} .

Saaty [14] démontre qu'une façon de déterminer les $w_{y_j}(z_i)$, c'est-à-dire les évaluations, $i = 1, \dots, m_{k+1}$ du niveau L_{k+1} consiste à travailler sur une matrice de comparaisons par paires comme indiqué au point II.1.2.

de cette façon, les $w_{y_j}(z_i), i = 1, \dots, m_{k+1}$ apparaissent pour chaque y_j fixé c'est-à-dire pour chaque critère, comme les composantes du vecteur propre normalisé correspondant à la valeur propre dominante de la matrice des comparaisons. Le même principe peut être appliqué pour déterminer les $w_x(y_j)$, c'est-à-dire les poids des critères.

II.2.3 Les étapes de la méthode.

1. Déterminer le vecteur \bar{W} des poids des critères à partir de la matrice des comparaisons par paires entre les critères.
2. Déterminer pour chaque critère i , les évaluations des actions, c'est-à-dire les $w_{y_j}(z_i) : i = 1, \dots, m_1$ à partir de la matrice des comparaisons par paires entre les actions.

On obtient ainsi la matrice B .

3. Calculer $W = B\bar{W}$; W est le vecteur correspondant au classement global des actions.

Remarque : L'analyse de sensibilité que l'on pourra réaliser portera uniquement sur la façon de construire la matrice des comparaisons par paires, c'est-à-dire :

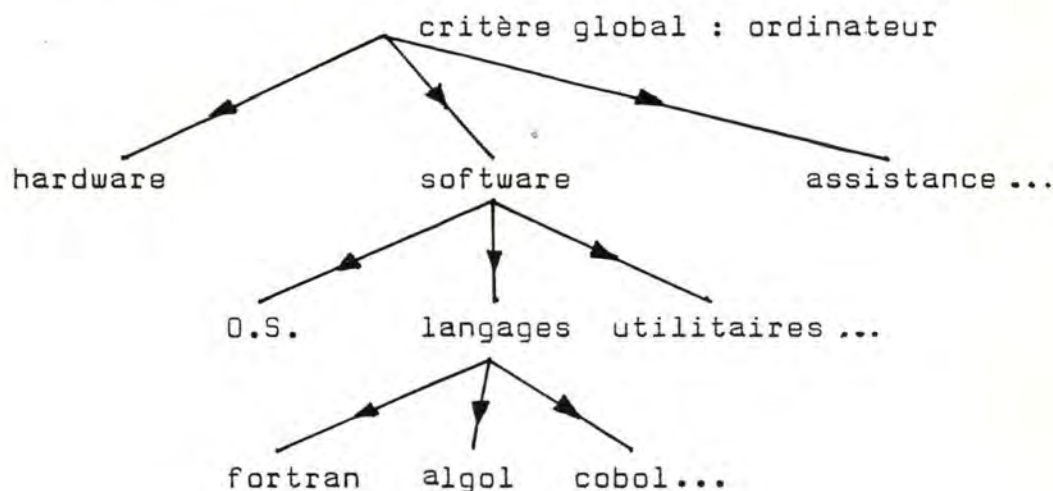
- en modifiant l'échelle choisie (point II.1.2.5)
- en modifiant les rapport p_{ij} (point II.1.2.5).

II.3 La méthode "MAL" de Dujmović [25], [23].

La méthode "MAL" (Mixed Averaging by Levels) en agrégeant les critères par la construction d'une fonction d'utilité doit permettre l'évaluation et la comparaison de "systèmes complexes". Par systèmes complexes, on entend un problème multicritère auquel est associé un "grand nombre" de critères, entre lesquels existent des interactions parfois difficiles à appréhender.

II.3.1 Présentation de la méthode.

La méthode "MAL" demande que l'on ait dressé une arborescence en éclatant les critères en sous-critères de plus en plus fins; par exemple :



Dans l'arborescence :

- les feuilles sont appelées les critères élémentaires (CE),
- les noeuds sont appelés critères non élémentaires (CNE),
- la racine est le critère global,
- un suivant d'un noeud est appelé un critère immédiatement surbordonné (CIS).

L'évaluation d'un CE se fait par l'intermédiaire d'une fonction d'efficacité g_i .

$$g_i : D_i \rightarrow I : u_i \rightarrow x_i = g_i(u_i)$$

où D_i est le domaine de définition du CE, et où $I = [0,1]$. Le nombre $x_i = g_i(u_i)$, appelé l'efficacité du CE de numéro i , représente le degré de vérité de la déclaration : "la valeur x_i satisfait complètement aux exigences du CE de numéro i ".

Cette fonction d'efficacité d'un CE fait intervenir plus qu'un ordre de préférence entre les actions pour ce CE, il faut donner une valeur exacte à chaque action que seules la méthode de construction d'une fonction d'utilité (cfr. point II.3) et la méthode des comparaisons par paires (cfr. point II.1.2) peuvent fournir si l'utilisateur n'en dispose pas.

II.3.2 Le principe de l'agrégation.

On évalue un CNE ayant n CE immédiatement subordonnés dont les efficacités respectives sont $x_1, x_2, \dots, x_n \in I$ par une fonction d'efficacité :

$$g : I^n \rightarrow I : (x_1, \dots, x_n) \rightarrow E = g(x_1, \dots, x_n).$$

Le nombre E est appelé l'efficacité du CNE, et représente le degré de vérité de la déclaration : "le CNE satisfait complètement aux exigences spécifiées".

On calcule ainsi de proche en proche les efficacités de tous les CNE et du critère global à partir de leurs CIS.

La fonction d'efficacité est :

$$g(x_1, \dots, x_n) = \begin{cases} \left(\sum_{i=1}^n p_i x_i^r \right)^{1/r} & \text{si } 0 < |r| < +\infty \\ \prod_{i=1}^n x_i^{p_i} & \text{si } r = 0 \\ \min(x_1, \dots, x_n) & \text{si } r = -\infty \\ \max(x_1, \dots, x_n) & \text{si } r = +\infty \\ 0 & \text{si } r < 0 \text{ et si } x_i = 0 \\ & \text{pour au moins un } i \in \{1, \dots, n\} . \end{cases}$$

Les p_i sont les poids accordés à chacun des n CIS, traduisant ainsi leur importance relative :

$$0 < p_i < 1 \text{ et } \sum_{i=1}^n p_i = 1.$$

r est déduit d'un degré de conjonction ou de disjonction choisi par l'utilisateur dans $[0, 1]$, à partir d'une table de 17 opérations logiques (cfr. tableau 1).

Dujmović justifie ces opérations logiques par ce qu'il appelle une "logique continue étendue" dérivant des moyennes pondérées :

$$M_n^{[r]}(x, p) = \lim_{s \rightarrow r} \left(\sum_{i=1}^n p_i \cdot x_i^s \right)^{1/s} ; r \in \bar{R} = R \cup \{-\infty, +\infty\}.$$

Tableau 1. Opérations logiques fondamentales de la logique continue étendue

A29.

Nom de l'opération logique	Symbole de l'opération	Degré de conjonction c	Degré de disjonction d	Paramètre r de la moyenne pondérée			
				n = 2	n = 3	n = 4	n = 5
Conjonction	C	1.0000	0.0000	- ∞	- ∞	- ∞	- ∞
Quasi-conjonction forte (+)	C ₊₊	0.9375	0.0625	-9.0600	-7.6393	-6.7100	-6.2723
Quasi-conjonction forte	C ₊	0.8750	0.1250	-3.5100	-3.1131	-2.8232	-2.6404
Quasi-conjonction forte (-)	C ₊₋	0.8125	0.1875	-1.6548	-1.5493	-1.4519	-1.3627
Quasi-conjonction moyenne	CA	0.7500	0.2500	-0.7203	-0.7314	-0.7145	-0.6697
Quasi-conjonction faible (+)	C ₋₊	0.6875	0.3125	-0.1478	-0.2065	-0.2248	-0.2025
Quasi-conjonction faible	C ₋	0.6250	0.3750	0.2612	0.1953	0.1678	0.1787
Quasi-conjonction faible (-)	C ₋₋	0.5625	0.4375	0.6194	0.5730	0.5473	0.5423
Indétermination conjonctive-disjonctive	A	0.5000	0.5000	1.0000	1.0000	1.0000	1.0000
Quasi-disjonction faible (-)	D ₋₋	0.4375	0.5625	1.4490	1.5190	1.5639	1.5838
Quasi-disjonction faible	D ₋	0.3750	0.6250	2.0185	2.1873	2.3019	2.3624
Quasi-disjonction faible (+)	D ₋₊	0.3125	0.6875	2.7917	3.1011	3.3181	3.4427
Quasi-disjonction moyenne	DA	0.2500	0.7500	3.9293	4.4501	4.8250	5.0542
Quasi-disjonction forte (-)	D ₊₋	0.1875	0.8125	5.8023	6.6748	7.3171	7.7294
Quasi-disjonction forte	D ₊	0.1250	0.8750	9.5207	11.0548	12.2766	13.0662
Quasi-disjonction forte (+)	D ₊₊	0.0625	0.9375	20.6303	24.3177	27.1346	29.0810
Disjonction	D	0.0000	1.0000	+ ∞	+ ∞	+ ∞	+ ∞

Posant :

$$M_n^{[r]}(x) = \lim_{s \rightarrow r} \left(\frac{1}{n} \cdot \sum_{i=1}^n x_i^s \right)^{1/s},$$

$$A_n = \int_0^1 dx_1 \int_0^1 dx_2 \dots \int_0^1 \max(x_1, \dots, x_n) dx_n = \frac{n}{n+1},$$

$$a_n = \int_0^1 dx_1 \int_0^1 dx_2 \dots \int_0^1 \min(x_1, \dots, x_n) dx_n = \frac{1}{n+1},$$

$$\bar{M}_n^{[r]} = \int_0^1 dx_1 \int_0^1 dx_2 \dots \int_0^1 M_n^{[r]}(x) dx_n$$

et remarquent que :

$$a_n \leq \bar{M}_n^{[r]} \leq A_n,$$

Dujmovic définit un degré de conjonction $c \in [0, 1]$:

$$c = \frac{A_n - \bar{M}_n^{[r]}}{A_n - a_n} = \frac{n - (n+1) \cdot \bar{M}_n^{[r]}}{(n-1)} = C_n(n) \quad n > 1,$$

qui détermine la position de $\bar{M}_n^{[r]}$ par rapport à la borne inférieure de l'intervalle $[a_n, A_n]$, et un degré de disjonction $d \in [0, 1]$:

$$d = \frac{\bar{M}_n^{[r]} - a_n}{A_n - a_n} = \frac{(n+1) \cdot \bar{M}_n^{[r]} - 1}{n-1} = D_n(r), \quad n > 1, \text{ qui}$$

détermine la position de $\bar{M}_n^{[r]}$ par rapport à la borne supérieure de l'intervalle $[a_n, A_n]$.

Par conséquent, pour un degré de conjonction \tilde{c} (ou de disjonction \tilde{d}),

$$\tilde{r}_n = C_n^{-1}(\tilde{c}) \text{ ou } \tilde{r}_n = D_n^{-1}(\tilde{d}) \text{ (formule 1).}$$

L'efficacité agrégée est alors calculée par $M_n^{[\tilde{r}_n]}(x, p)$.

Le calcul de \tilde{r}_n par la formule 1 étant compliqué, Dujmović met à la disposition de l'utilisateur la table des 17 opérations logiques qui donne la valeur de r pour des choix particuliers de \tilde{c} .

Le degré de conjonction c est interprété comme suit :

- si $c > .5$, ceci signifie que l'utilisateur désire

que l'ensemble des critères soit satisfait pour que l'efficacité du critère-père soit suffisante; en effet, r est alors inférieur à 1, et $Mn^{[r]}(x)$ sera plus près de $\min x_i$ que de $\max x_i$.

- si $c < .5$, l'utilisateur est moins exigeant quant à la concordance entre les efficacités des critères; pour une action, une grande efficacité d'un critère pourra compenser l'inefficacité d'un autre critère; en effet, r est alors supérieur à 1, et $Mn^{[r]}(x)$ sera plus près de $\max x_i$ que de $\min x_i$.
- si $c = d = 0.5$, alors $r = 1$: l'utilisateur ne prend position ni pour une concordance suffisante, ni pour une concordance faible entre les efficacités des critères.

Remarques :

1. r pourra faire l'objet d'une analyse de sensibilité.
2. une critique approfondie de la méthode est faite dans [25] .

II.4 La méthode des rangements médians.

II.4.1 Objectifs.

Si à chaque critère k , est associé un préordre P_k sur les actions comme expression des rapports de préférence partielle entre les actions, la recherche d'un rangement médian consiste à trouver, parmi l'ensemble B de tous les préordres globaux sur les actions, soit :

- un ou plusieurs préordres globaux P à distance minimale des k_1 préordres P_k
- un ou plusieurs préordres globaux P qui maximisent l'adhésion des k_1 préordres P_k à P .

Ces préordres globaux sont appelés médianes.

II.4.2 Intérêt de la méthode.

Lorsque l'utilisateur dispose, comme expression de l'importance relative de chaque critère, de sa position sur une échelle ordinale plutôt que d'un poids précis, l'obtention des médianes pour tous les systèmes de poids (p_1, \dots, p_{k_1}) possibles, qui respectent le préordre S associé à la position des critères sur l'échelle ordinale, est facilité par le fait qu'on peut discriminer ces systèmes de poids. En effet :

$$\text{les contraintes } \begin{cases} p_1 + \dots + p_{k_1} = 1 \\ p_1, p_2, \dots, p_{k_1} \geq 0 \end{cases}$$

définissent un polyèdre dans R^{k_1} dont les sommets sont $(1,0,\dots,0), (0,1,0,\dots,0), \dots, (0,0,\dots,1)$.

Les contraintes (1) définissent une partie réduite de ce polyèdre dont les sommets sont (2).

$$(1) \begin{cases} p_1 + \dots + p_{k_1} = 1 \\ p_1, \dots, p_{k_1} \geq 0 \\ p_k \geq p_{k_1} \quad k = 1, \dots, k_1 - 1 \end{cases} \text{ où les critères} \\ \text{sont numérotés selon l'ordre décroissant d'importance.}$$

$$(2) \begin{cases} (1,0,\dots,0) \\ (1/2,1/2,\dots,0) \\ \vdots \\ (1/k_1, 1/k_1, \dots, 1/k_1) \end{cases}$$

Si on considère un programme linéaire PL1 constitué d'une fonction objectif quelconque et des contraintes (1), les théorèmes de programmation linéaire [19] nous permettent de dire que :

1. Les solutions de PL1 sont à rechercher parmi les sommets définis en (2).
2. Si en 2 sommets distincts de (2), on obtient les mêmes médianes, alors pour tout système de poids appartenant à l'arête joignant les deux sommets, on obtiendra aussi les mêmes médianes.

Pour obtenir les médianes pour tous les systèmes de poids respectant le préordre S, il suffira, donc, de rechercher les médianes associées à chacun des sommets définis en (2).

II.4.3 Définition de distance et adhésion.

Si $A^1 = (a^1_{ij})$ et $B = (b_{ij})$ désignent les matrices d'adjacence associées respectivement au préordre P_1 et au préordre global P

$\{-p_k\}$ est un système de poids (p_1, \dots, p_{k_1}) ,

pour rechercher les médianes P à distance minimale des k_1 préordre P_1 , on utilise la distance de Kemeny définie comme suit [12] :

$$d = \sum_{l=1}^{k_1} p_l \sum_{i=1}^{m_1} \sum_{j=1}^{m_1} |a^1_{ij} - b_{ij}|.$$

La part de la paire d'actions (a, b) à l'éloignement d'un préordre P_1 par rapport à P vaut alors :

	aPb	aIb	bPa
aP_1b	0	1	2
aI_1b	1	0	1
bP_1a	2	1	0

(1)

Paelinck [26] définit l'adhésion à des k_1 préordres P_1 à un ordre P comme suit :

$$a = \sum_{l=1}^{k_1} p_l \times \sum_{i=1}^{m_1} \sum_{j=1}^{m_1} a^1_{ij} \times (2b_{ij} - 1), \text{ qu'il}$$

s'agit de maximiser.

La part de la paire d'actions (a,b) à l'adhésion de P_1 à P vaut :

	aPb	bPa
aP_1b	1	-1
aI_1b	0	0
bP_1a	-1	1

(2)

Si on veut définir l'adhésion a d'un préordre P_k à un préordre P , on définit, par extrapolation, la part de la paire d'actions (a,b) à l'adhésion de P_1 à P par :

	aPb	aIb	bPa
aP_1b	1	0	-1
aI_1b	0	1	0
bP_1a	-1	0	1

(3)

Remarques :

1. Si on compare les tableaux (1) et (3), on constate qu'ils sont équivalents à une constante (un) près et au signe près; par conséquent, les préordres médians définis par le calcul de d et de a seront les mêmes.
On s'attachera, par la suite, à la recherche de médiane en utilisant uniquement le concept de distance.

2. Plutôt que de vouloir minimiser l'éloignement global des préordres P_1 par rapport à P , on peut définir un autre objectif qui consiste à choisir P tel que, si P_e désigne le préordre, associé au critère e , pour lequel l'éloignement par rapport à P est maximum, cet éloignement soit minimum; c'est l'optique développée dans [28] .

II.4.4 Méthodes pour rechercher les préordres médians.

Le calcul de la distance d pour chaque préordre global possible, puis la sélection du(des) préordre(s) P à distance minimum devenant impossible lorsque le nombre d'actions devient élevé, il s'est avéré nécessaire de développer des méthodes plus efficaces.

La recherche d'une médiane peut ainsi être vue comme un problème de programmation linéaire en variables 0 - 1; les contraintes imposées portent sur les propriétés que le préordre médian P possède.

$$\left\{ \begin{array}{l} b_{ij} + b_{ji} \geq 1 \quad (P : \text{total et asymétrique}) \\ \forall i, j \in M \quad i \neq j \\ b_{ij} + b_{jk} - b_{ik} \leq 1 \quad (P : \text{transitif}) \quad \forall i, j, k \in M. \end{array} \right.$$

La résolution du P.L. PL2

$$\left\{ \begin{array}{l} [\min] \quad d \\ \text{sous les contraintes :} \\ b_{ij} + b_{ji} \geq 1, \forall i, j \in M \quad i \neq j \\ b_{ij} + b_{jk} - b_{ik} \leq 1, \forall i, j, k \in M. \quad (4) \end{array} \right.$$

devenant problématique lorsque le nombre d'actions est élevé, et par conséquent, la taille du tableau du simplexe associé à PL2 grande, nous avons choisi d'utiliser une technique "Branch and Bound" proposée par Cook et Saipé dans [27] .

II.4.5 Définitions et théorèmes préliminaires.

Si i et j sont deux entiers représentant deux actions $\in M$, avec $i < j$.

Définition 1 : i et j sont dits classés dans l'ordre naturel pour le critère k si $i I_k j$ ou $i P_k j$; si $j P_k i$, on dit qu'ils sont classés dans l'ordre non-naturel pour le critère k .

Définition 2 : on définit :

$$- r_{ij} \text{ où } r_{ij} = \sum_{l=1}^{k1} \delta_{ij}^l \text{ avec } \delta_{ij}^l = \begin{cases} p_l & \text{si } j P_l i \\ 0 & \text{sinon} \end{cases}$$

r_{ij} représente donc la somme des poids des critères pour lesquels i et j sont classés dans l'ordre non-naturel.

- m_{ij} où m_{ij} est la somme des poids des critères pour lesquels i et j ne sont pas indifférents.

Définition 3 : dans un préordre médian, seuls certaines actions peuvent être candidates pour être respectivement immédiatement préférées à, immédiatement préférées par, ou indifférentes à une action i . On définit ainsi :

- i est un candidat à être immédiatement préféré à j si dans tout préordre médian B dans lequel i précède immédiatement j, aucune amélioration de d ne peut être obtenue en rendant i indifférent ou préféré par j.
- i est un candidat à être préféré immédiatement par j si dans tout préordre médian B dans lequel j précède immédiatement i, aucune amélioration de d ne peut être obtenue en rendant i indifférent ou préféré à j.
- i est un candidat à être indifférent à j si dans tout préordre médian B dans lequel j est indifférent à i, aucune amélioration de d ne peut être obtenue en rendant i préféré immédiatement à ou par j.

Intuitivement, pour que i soit candidat à être immédiatement préféré à j, il faut que la somme des poids des critères pour lesquels i est préféré à j ne soit pas inférieure à la somme des poids restants, c'est-à-dire :

$$m_{ij} - r_{ij} \geq r_{ij} + (1 - m_{ij})$$

ou $m_{ij} - r_{ij} \geq 1/2.$

Si on applique un raisonnement similaire dans les cas où i est candidat à être immédiatement préféré ou indifférent à j, on obtient le théorème suivant.

Théorème 1 : pour deux actions i et j, i est candidat à être immédiatement préféré à j si $m_{ij} - r_{ij} \geq 1/2$; i est un candidat à être immédiatement préféré par j si $r_{ij} \geq 1/2$, et i est un candidat à être indifférent à j si $m_{ij} - 1/2 < r_{ij} < 1/2$.

Définition 4 : la matrice $\Gamma = (\gamma_{ij})$ avec $\gamma_{ii} = 0$ et $\gamma_{ij} = -\gamma_{ji}$ est définie comme suit :

$$\gamma_{ij} = \begin{cases} 1 & \text{si } m_{ij} - r_{ij} \geq 1/2 \\ 0 & \text{si } m_{ij} - 1/2 < r_{ij} < 1/2 \\ -1 & \text{si } r_{ij} \geq 1/2. \end{cases}$$

On démontre, de plus, que cette matrice Γ est telle que

$$\bar{B} = \sum_{l=1}^{k1} p_l \sum_{i=1}^{m1} \sum_{j=1}^{m1} |a^l_{ij} - \gamma_{ij}| \leq \sum_{l=1}^{k1} p_l \sum_{i=1}^{m1} \sum_{j=1}^{m1}$$

$$|a^l_{ij} - b_{ij}|, \forall \text{ préordre médian } B = (b_{ij}).$$

Remarque : Γ , telle que définie, est la matrice associée à une relation médiane quelconque. \bar{B} constitue de plus une limite inférieure à la distance d^* à rechercher.

Γ est donc la solution d'une relaxation du problème de recherche de préordre(s) médian(s) où la contrainte de transitivité (4) associée à PL2 a été abandonnée.

Les définitions et théorèmes suivants permettent de déterminer l'accroissement de \bar{B} dû au fait qu'on essaye de faire de Γ la représentation d'un préordre médian.

Définition 5 : on désigne par $P(i)$, $T(i)$ et $S(i)$ les candidats respectivement prédécesseurs, indifférents et successeurs de l'action i .

Théorème 2 : Γ correspond à un préordre médian si, et seulement si, les conditions de transitivité suivantes sont vérifiées $\forall i, j, k$.

$j \in S(i) \cup T(i)$ et $k \in S(j) \cup T(j) \Rightarrow k \in S(i) \cup T(i)$
avec $k \in T(i)$,
si, et seulement si, $j \in T(i)$ et $k \in T(j)$.

Théorème 3 :

$$\sum_{l=1}^{k1} d(A^l, \Gamma) = \sum_{(ij) \in \tilde{T}} (k1 - |m_{ij} - 2r_{ij}|) + \sum_{(ij) \in T} m_{ij}$$

où $T = \{(i, j) \mid i < j \text{ et } \gamma_{ij} = 0\}$ et $\tilde{T} = \{(i, j) \mid i < j \text{ et } \gamma_{ij} \neq 0\}$.

Théorème 4 :

si $\Gamma' = (\gamma'_{ij})$ où $\gamma'_{ij} = \gamma_{ij} \forall i, j$ sauf pour
 $\gamma'_{ab} = -\gamma_{ab}$ et $\gamma'_{ba} = -\gamma_{ba}$ ($\gamma_{ab} \neq 0$)
alors

$$\sum_{l=1}^{k1} d(A^l, \Gamma') = \sum_{l=1}^{k1} d(A^l, \Gamma) + 2 |m_{ab} - 2 r_{ab}|$$

Ce ~~théorème~~ permet de déterminer l'accroissement de d provoqué par le fait qu'on oblige b à appartenir à $S(a)$ (ou à $P(a)$) alors que d'après Γ , il appartenait à $P(a)$ (ou à $S(a)$).

Théorème 5 :

si $\Gamma' = (\gamma'_{ij})$ où $\gamma'_{ij} = \gamma_{ij}$ i, j sauf pour
 $\gamma'_{ab} = -1$ où $\gamma_{ab} = 0$

$$\text{alors } \sum_{l=1}^{k1} d(A^l, \Gamma') = \sum_{l=1}^{k1} d(A^l, \Gamma) + k1 - 2r_{ab}.$$

Ce théorème permet de déterminer l'accroissement de d provoqué par le fait qu'on oblige 2 sommets, qui étaient indifférents d'après Γ , à devenir préféré l'un par rapport à l'autre.

II.4.6 Principe de détermination du préordre médian.

Supposons :

1. qu'on ait obtenu un rangement partiel médian i_1, \dots, i_{p-1} , auquel est associée la borne inférieure \bar{M} sur d .
2. que l'on choisisse l'action i_p comme candidat indifférent à, ou préféré par, i_{p-1} dans le rangement partiel.

2 situations peuvent provoquer l'accroissement de \bar{M} .

1. Certaines actions i_k , qui, d'après Γ , devaient appartenir à $P(i_p)$, ne se trouvent pas dans le classement partiel i_1, \dots, i_{p-1} ; si l'on veut que Γ représente un préordre médian, ces actions doivent être "placées" dans $S(i_p)$, par conséquent, $\gamma_{i_k i_p}$ doit passer de 1 à -1; l'accroissement de \bar{M} est déterminé grâce au théorème 4.

2. Certaines actions i_k , qui d'après Γ , devaient être indifférentes à i_p , sont en fait dans le classement partiel i_1, \dots, i_{p-1} préférées à i_p ; pour que Γ représente un préordre médian, $\gamma_{i_p i_k}$ doit passer de 0 à -1. Le théorème 5 détermine l'accroissement de \bar{M} provoqué par cette modification de Γ .

Le principe du choix de l'action i_p est immédiat : on choisit, parmi les actions non encore classées, celle qui provoque l'accroissement minimum de \bar{M} .

Remarques :

1. Une solution partielle sera rejetée, si d'après Γ , à i_{p-1} ne sont associées que des actions qui lui sont préférées.
2. Si un rangement complet a été déterminé, on pourra rejeter un autre rangement, partiel celui-là, dès que la borne inférieure qui lui est associée est supérieure à la distance d associée au rangement complet.
3. Une fois un rangement complet déterminé, on recherche de nouveaux rangements en remplaçant l'action j occupant la position m_1-1 dans le rangement complet par celle qui provoque l'accroissement de la borne \bar{M} , associée au classement partiel i_1, \dots, i_{m_1-1} , immédiatement le plus faible après j . Si toutes les actions non classées en position i_{m_1-1} ont été envisagées pour le rangement partiel i_1, \dots, i_{m_1-2} , on considère la position m_1-2 dans le rangement complet, et on applique le même principe.

Le processus de recherche de rangement s'arrête lorsque toutes les actions ont été envisagées en position 1.

4. S'il existe une action $i_p \in S(i_{p-1}) \cup T(i_{p-1})$ tel que $P(i_p) \cup T(i_p) \subset \{i_1, \dots, i_{p-1}\}$,

alors la borne inférieure \bar{M} ne change pas si on ajoute i_p au rangement partiel $\{i_1, \dots, i_{p-1}\}$.

II.4.7 Algorithme.

Pour un système de poids $\{p_1, \dots, p_k\}$,

étape 1 : calculer $\{r_{ij}, i < j\}$, r , la borne initiale

$$M = \sum_{(ij) \in \tilde{T}} (m_1 - |m_{ij} - 2r_{ij}|) + \sum_{(ij) \in T} |m_{ij}| ; l = 0 .$$

Etape 2 : s'il existe une action i_1 tel que $P(i_1) = \emptyset$;
 $l = 1$; aller en 4.

Etape 3 : pour chaque $i \in \{1, \dots, m_1\}$, calculer la borne inférieure

$$\bar{M} = M + 2 \sum_{j \in P(i)} |m_{ij} - 2r_{ij}| ; \text{ soit } i_1$$

l'action à laquelle est associée la borne inférieure \bar{M} la plus faible ; $l = 1$.

Etape 4 :

soit : $\{i_1, \dots, i_l\}$ le rangement partiel avec la borne inférieure à la distance d égale à \bar{M} .

S'il existe un sommet $i_p \in S(i_1) \cup T(i_1)$ tel que $P(i_p) \cup T(i_p) \subset \{i_1, \dots, i_l\}$, ajouter i_p au classement partiel; $l = l + 1$.

Sinon : - calculer pour chaque i_p appartenant à $S(i_1) \cup T(i_1)$

$$\bar{N} = \bar{M} + 2 \sum_{j \in P(i_p) \setminus \{i_1, \dots, i_l\}} |m_{i_p j} - 2 r_{i_p j}|$$

$$+ \sum_{k \in T(i_p) \cap \{i_1, \dots, i_{l-2}\}} (m - 2 r_{k i_p})$$

- choisir l'action i_p à laquelle est associé le \bar{N} minimum;

$$\bar{M} = \bar{N}; \quad l = l + 1.$$

S'il n'existe pas de $i_p \in S(i_1) \cup T(i_1)$,

aller en 6.

Etape 5 :

- s'il existe déjà un rangement médian complet trouvé, auquel est associée la borne \bar{M}_c tel que $\bar{M}_c < \bar{M}$, rejeter le rangement $\{i_1, \dots, i_l\}$;

aller en 6.

- Si $l = m$, un rangement complet est déterminé, conserver ce rangement complet; poser $\overline{M}_c = M$; $l = l-1$; aller en 6.
- Sinon aller en 4.

Etape 6 : - si on a envisagé tous les rangements avec chaque action possible en position l , alors
 $l = l - 1$;
 si $l = 0$ stop,
 sinon aller en 6;
 - sinon, sélectionner l'action i_v à laquelle est associée la borne inférieure immédiatement inférieure à celle de i_1 ; mettre i_v à la place de i_1 dans le rangement partiel $\{i_1, \dots, i_1\}$;
 aller en 4.

Annexe III : description des fichiers utilisés.

La description des fichiers utilisés est divisée en deux grandes parties :

- la première partie est consacrée à la description des fichiers qui enregistrent, pour chaque type d'outil tel que décrit au chapitre II, chacun des rapports de préférence déterminé à partir de cet outil;
- la seconde partie décrit les fichiers qui enregistrent les paramètres et les résultats propres à chacune des méthodes multicritères.

A chaque fichier est assigné un numéro d'unité logique compris entre 0 et 99. Fortran ne permettant pas que deux fichiers, auxquels sont associés les mêmes numéros d'unité logique, soient ouverts en même temps, nous avons réservé les numéros 66 à 79 pour les fichiers propres à chaque méthode multicritère ; hormis les numéros :

- 5, utilisé pour la lecture au terminal
- 6, utilisé pour l'écriture au terminal
- 3 réservé pour l'impression sur la grosse imprimante,

ainsi que 47, 80, 81, les autres numéros peuvent être assignés aux fichiers enregistrant les rapports de préférence.

Pour chaque fichier décrit, on retrouve la structure suivante :

- le nom : ****.DAT, codé sur 30 caractères maximum;

- le nom et le format de chaque enregistrement constituant le fichier;
- la clef d'accès, c'est-à-dire la liste des enregistrements qui permet l'accès à un article donné du fichier.

Ainsi si

enreg1	(I1)	}	clef d'accès
enreg2	(I2)		
enreg3	(I2)		

sont désignés comme étant les enregistrements constituant la clef d'accès, celle-ci sera établie de la façon suivante :

clef d'accès = 1000xenreg1 + 100xenreg2 + enreg3;

- le numéro d'unité logique;
- un commentaire facultatif sur le fichier décrit.

Les fichiers de base sont ceux relatifs aux notions de problèmes multicritères, d'actions et de critères. On définit ainsi :

PROBLEME.DAT

- numéro du problème multicritère (I2)
 - le nom de ce problème multicritère (A30)
 - le nombre d'actions (I2)
 - le nombre de critères (I2).
-] clef d'accès

Numéro-unité : 81.

Commentaire : on référencera, par la suite, le numéro du problème multicritère par nmprob.

ACTION.DAT

- nmprob (I2)
 - numéro d'ordre (I2)
 - le nom de cette action (A30).
- } clef d'accès

Numéro-unité : 80.

Commentaire : ce fichier permet d'enregistrer le nom de
chaque action.

CRITERE.DAT

- nmprob (I2)
 - numéro d'ordre (I2)
 - le nom de ce critère (A30).
- } clef d'accès

Numéro-unité : 47.

Commentaire : ce fichier permet d'enregistrer le nom de
chaque critère.

III.1 L'enregistrement des rapports de préférence.

III.1.1 L'enregistrement des relations de préférence.

A chaque rapport de préférence, sont associés 3 fichiers correspondant aux 3 types de relations possibles : relation quelconque, quasi-ordre et préordre.

Lorsque la relation introduite par l'utilisateur est une relation quelconque, on enregistre sa matrice d'adjacence; si c'est un quasi-ordre, on enregistre la "position" de chaque action ou de chaque critère telle qu'elle est définie en Annexe I; si c'est un préordre, on enregistre le nombre - rentré par l'utilisateur ou déduit du préordre -, associé à chaque action ou à chaque critère, qui traduit sa position dans le préordre; plus ce nombre est élevé, et plus le critère ou l'action correspondant est mieux classé.

Si la relation introduite par l'utilisateur est quelconque, ou est un quasi-ordre, on doit pouvoir retrouver le préordre finalement choisi par l'utilisateur; de même, on doit pouvoir retrouver la relation construite à partir du préordre choisi par l'utilisateur.

De plus, pour un même quasi-ordre, ou une même relation quelconque, construit, l'utilisateur peut avoir choisi, au cours d'analyses de sensibilité successives, des préordres distincts; de même, l'utilisateur peut choisir un même préordre au départ de relations quelconques, ou de quasi-ordres, construits qui sont distincts.

Les fichiers suivants ont, par conséquent, été définis.

III.1.1.1 Les fichiers associés aux rapports de préférence partielle entre les actions.

RELATION-ACT-CRT.DAT

- nmprob	(I2)	} clef d'accès
- numéro-critère	(I2)	
- numéro-ordre-relation	(I2).	

Numéro-unité : 15.

Commentaire : un article du fichier correspond à l'existence d'une relation quelconque construite par l'utilisateur.

RAPPORT-REL-ACT-CRT.DAT

- nmprob	(I2)	} clef d'accès
- numéro-critère	(I2)	
- numéro-ordre-relation	(I2)	
- num-action-ligne	(I2)	
- num-action-colonne	(I2)	
- valeur-élément-adjacent	(I1).	

Numéro-unité : 11.

Commentaire : ce fichier permet l'enregistrement de la matrice d'adjacence.

QUASI-ORDRE-ACT-CRT.DAT

- nmprob (I2)
- numéro-critère (I2)
- numéro-ordre-quasi-ordre (I2).

} clef d'accès

Numéro-unité : 34.

Commentaire : un article du fichier correspond à l'existence d'un quasi-ordre.

POSITION-QO-ACT-CRT.DAT

- nmprob (I2)
- numéro-critère (I2)
- numéro-ordre-quasi-ordre (I2)
- numéro-action (I2)
- position-action (F5.1).

} clef d'accès

Numéro-unité : 35.

PREORDRE-ACTION-CRITERE.DAT

- nmprob (I2)
- numéro-critère (I2)
- numéro-ordre-préordre (I2).

} clef d'accès

Numéro-unité : 82.

Commentaire : un article correspond à l'existence d'un préordre total.

RANG-ACT-CRT.DAT

- nmprob (I2)
- numéro-critère (I2)
- numéro-ordre-préordre (I2)
- num-action (I2)
- rang-action (F9.2).

} clef d'accès

Numéro-unité : 7.

PREORDRE-REL-QO-ACT-CRT.DAT

- nmprob (I2)
- numéro-critère (I2)
- numéro-ordre-préordre (I2)
- numéro-ordre (I2)
- numéro-ordre-relation/quasi-ordre (I2)
- type-relation (I1).

} clef d'accès

Numéro-unité : 33.

Commentaire : ce fichier permet de retrouver, à partir du préordre choisi par l'utilisateur, la (les) relation(s) quelconque(s) (si type-relation = 1) ou le(les) quasi-ordre(s) (si type-relation = 2) qu'il avait tout d'abord construit.

REL-QO-PREORDRE-ACT-CRT.DAT

- nmprob (I2)	}	clef d'accès
- numéro-critère (I2)		
- numéro-ordre-relation/quasi-ordre (I2)		
- type-relation (I1)		
- numéro-ordre (I2)		
- numéro-ordre-préordre (I2),		

Numéro-unité : 12.

Commentaire : ce fichier permet de retrouver le(s) pré-ordre(s) finalement choisi(s) par l'utilisateur, à partir de la relation quelconque (si type-relation = 1), ou du quasi-ordre (si type-relation = 2) qu'il avait au préalable construit.

III.1.1.2 Les fichiers associés aux rapports de préférence entre les critères.

RELATION-CRITERE.DAT

- nmprob (I2)
 - numéro-ordre-relation (I2).
- } clef d'accès

Numéro-unité : 20.

Commentaire : un article du fichier correspond à l'existence d'une relation quelconque construite par l'utilisateur.

RAPPORT-REL-CRT.DAT

- nmprob (I2)
 - numéro-ordre-relation (I2)
 - num-critère-ligne (I2)
 - num-critère-colonne (I2)
 - valeur-élément-adjacent (I1).
- } clef d'accès

Numéro-unité : 24.

Commentaire : ce fichier permet l'enregistrement de la matrice d'adjacence.

QUASI-ORDRE-CRT.DAT

- nmprob (I2)
 - numéro-ordre-quasi-ordre (I2).
- } clef d'accès

Numéro-unité : 21.

Commentaire : un article du fichier correspond à l'existence d'un quasi-ordre construit par l'utilisateur.

POSITION-QO-CRT.DAT

- nmprob (I2)	}	clef d'accès
- numéro-ordre-quasi-ordre (I2)		
- numéro-critère (I2)		
- position-critère (F5.1).		

Numéro-unité : 23.

PREORDRE-CRITERE.DAT

- nmprob (I2)	}	clef d'accès
- numéro-ordre-préordre (I2).		

Numéro-unité : 18.

Commentaire : un article du fichier correspond à un pré-ordre total choisi ou construit par l'utilisateur.

RANG-CRITERE.DAT

- nmprob (I2)	}	clef d'accès
- numéro-ordre-préordre (I2)		
- numéro-critère (I2)		

- rang-critère (F4.1).

Numéro-unité : 17.

Commentaire : le vecteur des rangs des critères est normalisé sur dix.

PREORDRE-REL-QO-CRT.DAT

- nmprob (I2)
 - numéro-ordre-préordre (I2)
 - numéro-ordre (I2)
 - numéro-ordre-relation/quasi-ordre (I2)
 - type-relation (I1).
- } clef d'accès

Numéro-unité : 27.

Commentaire : ce fichier permet de retrouver, à partir du préordre choisi par l'utilisateur, la (les) relation(s) quelconque(s) (si type-relation = 1), ou le(s) quasi-ordre(s) (si type-relation = 2) qu'il avait tout d'abord construit avant de choisir ce préordre.

REL-QO-PREORDRE-ACT-CRT.DAT

- nmprob (I2)
 - numéro-ordre-relation/quasi-ordre (I2)
 - type-relation (I1)
 - numéro-ordre (I2)
 - numéro-ordre-préordre (I2).
- } clef d'accès

Numéro-unité : 26.

Commentaire : ce fichier permet de retrouver le(s) pré-ordre(s) finalement choisi(s) par l'utilisateur, à partir de la relation quelconque (si type-relation = 1), ou du quasi-ordre (si type-relation = 2) qu'il avait au préalable construit.

III.1.1.3 Les fichiers associés aux rapports de préférence globale entre les actions et aux relations de surclassement déterministes.

Les fichiers suivants sont non seulement utilisés pour enregistrer les relations de préférence globale construites par l'utilisateur, mais aussi les relations de préférence globale déterminées par les méthodes multicritères d'agrégation.

RELATION-ACTION-GLOBALE.DAT

- nmprob (I2)	}	clef d'accès
- numéro-ordre-relation (I2).		

Numéro-unité : 51.

Commentaire : un article d'un fichier correspond à l'existence d'une relation quelconque construite par l'utilisateur, ou d'une relation de préférence globale, déterminée par une méthode d'agrégation, qui est une relation quelconque.

RAPPORT-REL-ACTGL.DAT

- nmprob (I2)
- numéro-ordre-relation (I2)
- num-action-ligne (I2)
- num-action-colonne (I2)
- valeur-élément-adjacent (I1)

}
clef d'accès

Numéro-unité : 52.

Commentaire : ce fichier permet l'enregistrement de la matrice d'adjacence.

QO-ACT-GLOBALE.DAT

- nmprob (I2)
- numéro-ordre-quasi-ordre (I2)

}
clef d'accès

Numéro-unité : 53.

Commentaire : un article du fichier correspond à l'existence d'un quasi-ordre construit par l'utilisateur, ou d'une relation de préférence globale, déterminée par une méthode d'agrégation, qui est un quasi-ordre.

POSITION-QO-CRT.DAT

- nmprob (I2)
- numéro-ordre-quasi-ordre (I2)
- numéro-action (I2)
- position-action (F5.1).

}
clef d'accès

Numéro-unité : 54.

PREORDRE-ACTION-GLOBAL.DAT

- | | | |
|-------------------------------|---|--------------|
| - nmprob (I2) | } | clef d'accès |
| - numéro-ordre-préordre (I2). | | |

Numéro-unité : 50.

Commentaire : un article du fichier correspond à l'existence d'un préordre construit ou rentré par l'utilisateur, ou d'une relation de préférence globale, déterminée par une méthode d'agrégation, qui est un préordre.

RANG-PREORDRE-ACTGL.DAT

- | | | |
|------------------------------|---|--------------|
| - nmprob (I2) | } | clef d'accès |
| - numéro-ordre-préordre (I2) | | |
| - numéro-action (I2) | | |
| - rang-moyen-action (F4.1). | | |

Numéro-unité : 43.

PREORDRE-REL-QO-ACTGL.DAT

- | | | |
|--|---|--------------|
| - nmprob (I2) | } | clef d'accès |
| - numéro-ordre-préordre (I2) | | |
| - numéro-ordre (I2) | | |
| - numéro-ordre-relation/quasi-ordre (I2) | | |
| - type-relation (I1). | | |

Numéro-unité : 58.

Commentaire : ce fichier permet de retrouver, à partir du préordre choisi par l'utilisateur, le (les) relation(s) quelconque(s) (si type-relation = 1) ou le(s) quasi-ordre(s) (si type-relation = 2) qu'il avait tout d'abord construit avant de choisir ce préordre.

REL-QO-PREORDRE-ACTGL.DAT

- nmprob (I2)	}	clef d'accès
- numéro-ordre-relation/quasi-ordre (I2)		
- type-relation (I1)		
- numéro-ordre (I2)		
- numéro-ordre-préordre (I2)		

Numéro-unité : 57.

Commentaire : ce fichier permet de retrouver le(les) préordre(s) finalement choisi(s) par l'utilisateur, à partir de la relation quelconque (si type-relation = 1), ou du quasi-ordre (si type-relation = 2) qu'il avait au préalable construit.

III.1.2 Les fichiers associés à la méthode de rangement par des comparaisons par paires de Saaty.

A partir des considérations du point IV.1.2.3.3, on a retenu, pour chacun des 3 rapports de préférences possibles, les 2 fichiers suivants :

1. un fichier qui reprend :

- l'échelle choisie pour construire la matrice des comparaisons par paires,
- la valeur propre dominante associée à cette matrice,
- le numéro du préordre correspondant au vecteur propre associé à la valeur propre dominante.

2. un fichier qui enregistre la matrice des comparaisons par paires.

De plus, pour chacun des rapports de préférence, on définit un fichier qui, à partir du préordre finalement choisi par l'utilisateur, permettra de retrouver la(les) matrice(s) des comparaisons par paires construite(s) si la méthode de rangement a été utilisée pour déterminer les rapports de préférence.

Les fichiers suivants ont, par conséquent, été définis.

III.1.2.1 Les fichiers associés aux rapports de préférence partielle entre les actions.

METH-SAATY-PREORDRE-ACT.DAT

- nmprob (I2)
- numéro-critère (I2)
- numéro-ordre-méthode (I2)
- valeur-propre (F7.2)
- échelle-choisie (I1)
- numéro-préordre (I2).

} clef d'accès

Numéro-unité : 34.

RAPPORT-ACT-CRT-SAATY.DAT

- nmprob (I2)	}	clef d'accès
- numéro-critère (I2)		
- numéro-ordre-méthode (I2)		
- numéro-action-ligne (I2)		
- numéro-action-colonne (I2)		
- rapport (F4.2).		

Numéro-unité : 35.

Commentaire : ce fichier permet l'enregistrement de la
matrice des comparaisons par paires.

RAP-PREORDRE-ACT-CRT-SAATY.DAT

- nmprob (I2)	}	clef d'accès
- numéro-critère (I2)		
- numéro-préordre-choisi-par-l'utilisateur (I2)		
- numéro-ordre (I2)		
- numéro-ordre-méthode (I2).		

Numéro-unité : 36.

Commentaire : ce fichier permet, à partir du numéro du
préordre choisi par l'utilisateur, de re-
trouver le(s) numéro(s) d'ordre de la (des)
méthode(s) de rangement utilisée(s) qui
a(ont) abouti au choix, par l'utilisateur,
de ce préordre.

III.1.2.2 Les fichiers associés aux rapports de préférence entre les critères.

SAATY-PREORDRE-CRT.DAT

- nmprob (I2)
- numéro-ordre-méthode (I2)
- valeur propre (F7.2)
- échelle choisie (I1)
- numéro-préordre (I2).

}

clef d'accès

Numéro-unité : 43.

RAPPORT-CRT-SAATY.DAT

- nmprob (I2)
- numéro-ordre-méthode (I2)
- numéro-critère-ligne (I2)
- numéro-critère-colonne (I2)
- rapport (F4.2).

}

clef d'accès

Numéro-unité : 44.

Commentaire : ce fichier permet l'enregistrement de la matrice des comparaisons par paires.

RAP-PREORDRE-CRT-SAATY.DAT

- nmprob (I2)
 - numéro-préordre-choisi-par-l'utilisateur (I2)
 - numéro-ordre (I2)
 - numéro-ordre-méthode (I2).
- } clef
d'accès

Numéro-unité : 45.

Commentaire : ce fichier permet de retrouver le(s) numéro(s) d'ordre de la(des) méthode(s) utilisée(s), à partir de laquelle (desquelles) l'utilisateur a choisi le préordre pour lequel il dispose du numéro.

III.1.2.3 Les fichiers associés aux rapports de préférence globale entre les actions.

SAATY-ACTGL.DAT

- nmprob (I2)
 - numéro-ordre-méthode (I2)
 - valeur propre (F7.2)
 - échelle choisie (I1)
 - numéro-préordre (I2).
- } clef d'accès

Numéro-unité : 61.

RAP-SAATY-ACTGL.DAT

- nmprob (I2)
 - numéro-ordre-méthode (I2)
 - numéro-action-ligne (I2)
 - numéro-action-colonne (I2)
 - rapport (F4.2).
- } clef d'accès

Numéro-unité : 62.

Commentaire : ce fichier permet l'enregistrement de la
matrice des comparaisons par paires.

RAP-PREORDRE-ACTGL-SAATY.DAT

- nmprob (I2)
 - numéro-préordre-choisi-par-l'utilisateur (I2)
 - numéro-ordre (I2)
 - numéro-ordre-méthode (I2).
- } clef d'accès

Numéro-unité : 63.

Commentaire : ce fichier permet, à partir du numéro du
préordre choisi par l'utilisateur, de re-
trouver le(s) numéro(s) d'ordre de la (des)
méthode(s) de rangement utilisée(s) qui a
(ont) abouti au choix, par l'utilisateur
de ce préordre.

III.1.3 Les fichiers associés à la construction d'une fonction d'utilité pour déterminer les rapports de préférence partielle entre les actions.

Les fonctions d'utilité considérées sont :

- celles construites par l'utilisateur à l'aide de la méthode du point de valeur médiane (cfr. chpt. II.3);
- celles construites par la méthode UTA I (cfr. chpt. III.3.2).

Les fichiers définis sont les suivants :

FONCTION-UTILITE.DAT

- | | | |
|---|---|--------------|
| <ul style="list-style-type: none"> - nmprob (I2) - num-critère (I2) - num-ordre-fonction-utilité (I2). | } | clef d'accès |
|---|---|--------------|

Numéro-unité : 37.

Commentaire : un article du fichier correspond à l'existence d'une fonction d'utilité pour un critère donné.

CRT-ACTION-REPERE.DAT

- | | | |
|---|---|--------------|
| <ul style="list-style-type: none"> - nmprob (I2) - num-critère (I2) - num-ordre-fonction-utilité (I2) - num-ordre-point-estimé (I2) | } | clef d'accès |
|---|---|--------------|

- abscisse du point estimé (F9.2)
- utilité du point repère (F4.2).

Numéro-unité : 39.

Commentaire : un article du fichier correspond à l'existence d'un point estimé de la fonction d'utilité, caractérisé par son abscisse et son utilité.

RAP-PREORDRE-ACT-CRT-FCT-UTILITE.DAT

- | | | |
|------------------------------------|---|--------------|
| - nmprob (I2) | } | clef d'accès |
| - num-critère (I2) | | |
| - num-préordre (I2) | | |
| - num-ordre (I2) | | |
| - num-ordre-fonction-utilité (I2). | | |

Numéro-unité : 40.

Commentaire : ce fichier permet de retrouver le(s) numéro(s) de la (des) fonction(s) d'utilité à partir de laquelle (desquelles) l'utilisateur a choisi le préordre pour lequel il dispose du numéro.

Ce fichier est utilisé uniquement pour les fonctions d'utilité construites à l'aide de la méthode du point de valeur médiane.

REFERENCE-ACT-CRT.DAT

- nmprob (I2)
 - numéro-critère (I2)
 - numéro d'une action (I2)
 - abscisse de cette fonction (F9.2).
- } clef d'accès

Numéro-unité : 48.

Commentaire : ce fichier enregistre l'abscisse de chacune des actions d'un problème multicritère pour un critère.

BORNE-MESURE-CRT.DAT

- nmprob (I2)
 - numéro-critère (I2)
 - borne inférieure de la mesure d'un critère (F9.2)
 - borne supérieure de la mesure d'un critère (F9.2).
- } clef d'accès

Numéro-unité : 49.

Commentaire : ce fichier enregistre les bornes gi_* et gi^* de variation de la mesure d'un critère; les abscisses de toutes les actions pour ce critère seront comprises entre ces bornes.

III.2 L'enregistrement des paramètres et résultats associés à chaque méthode multicritère.

Pour chaque méthode multicritère, le numéro d'unité logique 66 est réservé à un fichier qui enregistre les éléments propres à chaque analyse de sensibilité : son nom, ainsi que les valeurs des paramètres et résultats qui sont définis au niveau de la méthode, et non au niveau de chaque critère; le numéro d'unité logique 74 est réservé à un fichier qui enregistre les éléments définis au niveau de chaque critère.

On désignera dans la description des fichiers :

- le numéro de préordre entre les critères par nmpcr (ce préordre traduit les rapports de préférence entre les critères).
- le numéro du préordre entre les actions pour un critère par nmpact (ce préordre traduit les rapports de préférence partielle entre les actions pour un critère).

III.2.1 Les fichiers associés à la méthode Oreste.

METHODE-ORESTE.DAT

- | | | |
|--|---|--------------|
| - nmp _{prob} (I2) | } | clef d'accès |
| - numéro-analyse-sensibilité (I2) | | |
| - nmp _{pcr} (I2) | | |
| - la distance choisie (I1) | | |
| - le paramètre associé à cette distance (F5.2) | | |
| - le seuil d'indifférence (F5.4) | | |
| - le seuil d'incomparabilité (F7.4) | | |
| - la méthode extérieure choisie (I1) | | |
| - le nom de cette analyse de sensibilité (A30) | | |
| - le numéro de la relation de surclassement (I2) | | |
| - le type de cette relation de surclassement (I1). | | |

Numéro-unité : 66.

Commentaire : un article de ce fichier correspond à une analyse de sensibilité à laquelle on a appliqué la méthode Oreste.

Le type de la relation de surclassement vaut :

- 1, si elle est quelconque
 - 2, si elle est un quasi-ordre
 - 3, si elle est un préordre.
-

ORESTE-QO-APPROXIME.DAT

- nmprob (I2)
 - numéro-analyse-sensibilité (I2)
 - numéro du quasi-ordre , approximation
de la relation de surclassement,
 - discriminant (I2)
 - non-discriminant (I2)
 - intermédiaire (I2).
- } clef d'accès

Numéro-unité : 67.

Commentaire : ce fichier permet d'enregistrer, lorsque l'utilisateur demande une approximation de la relation de surclassement, les numéros des 3 quasi-ordres.

PREORDRE-ACT-CRT-ORESTE.DAT

- nmprob (I2)
 - numéro-analyse-sensibilité (I2)
 - numéro-critère (I2)
 - nmpact (I2).
- } clef d'accès

Numéro-unité : 74.

Commentaire : ce fichier enregistre les paramètres définis pour chaque critère de l'analyse de sensibilité.

III.2.2 Les fichiers associés à la méthode Electre I.

METHODE-ELECTRE-UN.DAT

- | | | |
|--|---|--------------|
| - nmprob (I2) | } | clef d'accès |
| - numéro-analyse-sensibilité (I2) | | |
| - nmpcr (I2) | | |
| - seuil de concordance (F4.2) | | |
| - test de concordance (I1) | | |
| - le nom de cette analyse de sensibilité (A30) | | |
| - le numéro de la relation de surclassement (I2) | | |
| - le type de cette relation de surclassement (I1). | | |

Numéro-unité : 66.

Commentaire : un article de ce fichier correspond à une analyse de sensibilité à laquelle on a appliqué la méthode Electre I.
Le type de la relation de surclassement vaut :

1. si elle est quelconque
2. si elle est un quasi-ordre
3. si elle est un préordre.

PREORDRE-ACT-CRT-ELUN.DAT

- | | | |
|-----------------------------------|---|--------------|
| - nmprob (I2) | } | clef d'accès |
| - numéro-analyse-sensibilité (I2) | | |
| - numéro-critère (I2) | | |
| - nmpact (I2) | | |
| - indice-discordance (F9.2). | | |

Numéro-unité : 74.

Commentaire : ce fichier enregistre les paramètres définis pour chaque critère de l'analyse de sensibilité.

III.2.3 Les fichiers utilisés par les méthodes Oreste et Electre I.

Les fichiers qui sont communs aux deux méthodes sont ceux qui enregistrent les quasi-noyaux et les circuits.

QUASI-NOYAU.DAT

- | | | |
|---|---|--------------|
| - nmprob (I2) | } | clef d'accès |
| - numéro-analyse-sensibilité (I2) | | |
| - num-méthode (I1) | | |
| - le numéro d'ordre du quasi-noyau (I2) | | |
| - la faiblesse de ce quasi-noyau (I2). | | |

Numéro-unité : 77.

Commentaire : ce fichier permet d'enregistrer l'existence d'un quasi-noyau pour une relation de surclassement associée à l'exécution de la méthode Oreste ou Electre I sur une analyse de sensibilité donnée.

Si num-méthode égale 1, c'est la méthode Oreste qui a été exécutée; si num-méthode égale 2, c'est la méthode Electre I.

QUASI-N-ACTION.DAT

- nmprob (I2)
- numéro-analyse-sensibilité (I2)
- num-méthode (I1)
- le numéro d'ordre du quasi-noyau (I2)
- le numéro d'ordre de l'action (I2)
- appartenance-action (I1).

}
clef d'accès

Numéro-unité : 69.

Commentaire : ce fichier enregistre pour chaque action d'un quasi-noyau, à quel sous-ensemble elle appartient; ainsi, si :

- appartenance-action vaut 1, l'action appartient au quasi-noyau,
- appartenance-action vaut 2, l'action appartient au sous-ensemble des "dominés",
- appartenance-action égale 3, l'action appartient au sous-ensemble des "quasi-dominés".

CIRCUIT.DAT

- nmprob (I2)
- numéro-analyse-sensibilité (I2)
- num-méthode (I1)
- le numéro d'ordre du circuit (I2).

}
clef d'accès

Numéro-unité : 70.

Commentaire : un article de ce fichier correspond à l'existence d'un circuit dans la relation de sur-

classement déterminée par Oreste (si num-
méthode égale 1) ou par Electre I (si num-
méthode vaut 2).

CIRCUIT-ACTION.DAT

- | | | |
|--|---|--------------|
| <ul style="list-style-type: none"> - nmprob (I2) - numéro-analyse-sensibilité (I2) - num-méthode (I1) - le numéro d'ordre du circuit (I2) - un numéro d'ordre (I2) - le numéro d'une action appartenant
à ce circuit (I2). | } | clef d'accès |
|--|---|--------------|

Numéro-unité : 71.

Commentaire : un article de ce fichier enregistre le nu-
méro d'une action appartenant à ce circuit.

III.2.4 Les fichiers associés à la méthode Electre-II.

METHODE-ELDEU.DAT

- | | | |
|--|---|--------------|
| <ul style="list-style-type: none"> - nmprob (I2) - numéro-analyse-sensibilité (I2) - seuil de concordance C1 (F4.2) - seuil de concordance C2 (F4.2) - seuil de concordance C3 (F4.2) - nmpr(I2) - test de concordance (I1) | } | clef d'accès |
|--|---|--------------|

- le nom de cette analyse de sensibilité (A30)
- le numéro de la relation de surclassement forte (I2)
- le type de la relation de surclassement forte (I1)
- le numéro de la relation de surclassement faible (I2)
- le type de la relation de surclassement faible (I2).

Numéro-unité : 66.

Commentaire : un article de ce fichier correspond à une analyse de sensibilité à laquelle on a appliqué la méthode Electre II.

PREORDRE-GLOBAL-ACT-ELDEU.DAT

- | | |
|---|----------------|
| - nmprob (I2) | } clef d'accès |
| - numéro-analyse-sensibilité (I2) | |
| - numéro-classement-direct (I2) | |
| - numéro-classement-indirect (I2) | |
| - coefficient de corrélation entre les 2 classements (55,3) | |

Numéro-unité : 72.

Commentaire : lorsque l'utilisateur demande la détermination des classements, ce fichier enregistre les numéros des préordres globaux correspondant à ces classements, ainsi que le coefficient de corrélation entre les classements directs et indirects.

PREORDRE-ACT-CRT-ELDEU.DAT

- nmprob (I2)
 - numéro-analyse-sensibilité (I2)
 - numéro-critère (I2)
 - nmpact (I2)
 - seuil de discordance d1i (F9.2)
 - seuil de discordance d2i (F9.2).
- } clef d'accès

Numéro-unité : 74.

III.2.5 Les fichiers associés à la méthode Electre III.METHODE-ELTROIS.DAT

- nmprob (I2)
 - numéro-analyse-sensibilité (I2)
 - nmpcr (I2)
 - le nom de cette analyse de sensibilité (A30).
- } clef d'accès

Numéro-unité : 66.

Commentaire : un article de ce fichier correspond à une analyse de sensibilité à laquelle on a appliqué la méthode Electre III.

ELTROIS-ACTION.DAT

- nmprob (I2)
 - numéro-analyse-sensibilité (I2)
- } clef d'accès

- élément constant du seuil de discrimination (F4.2)
- élément variable du seuil de discrimination (F5.2)
- numéro-classement-descendant (I2)
- numéro-classement-ascendant (I2)
- coefficient de corrélation entre les 2 classements (F5.3)

Numéro-unité : 67.

Commentaire : lorsque l'utilisateur demande la détermination des classements, ce fichier enregistre les numéros des préordres globaux correspondant à ces classements, le coefficient de corrélation entre les classements ascendants et descendants, et le seuil de discrimination.

SURCLASSEMENT-ELTROIS.DAT

- | | | |
|--|---|--------------|
| <ul style="list-style-type: none">- nmprob (I2)- numéro-analyse-sensibilité (I2)- num-action-ligne (I2)- num-action-colonne (I2)- degré de crédibilité (F4.2). | } | clef d'accès |
|--|---|--------------|

Numéro-unité : 68.

Commentaire : pour chaque action, ce fichier enregistre la crédibilité de son surclassement sur chacune des autres actions.

ELTROIIS-PREORDRE-ACT-CRT.DAT

- nmprob (I2)
- numéro-analyse-sensibilité (I2)
- numéro-critère (I2)
- nmpact (I2)
- élément constant du seuil de préférence (F5.2)
- élément variable du seuil de préférence (F9.2)
- élément constant du seuil d'indifférence (F5.2)
- élément variable du seuil d'indifférence (F9.2)
- élément constant du seuil de veto (F5.2)
- élément variable du seuil de veto (F9.2).

}

clef d'accès

Numéro-unité : 74.III.2.6 Les fichiers associés à la méthode hiérarchique des comparaisons par paires de Saaty.

Cette méthode utilise les fichiers, définis aux points III.1.2.1 et III.1.2.2 de cette annexe, qui enregistrent les rapports de préférence partielle entre les actions et les rapports de préférence entre les critères déterminés à partir de la méthode de rangement par des comparaisons par paires de Saaty.

METHODE-SAATY.DAT

- nmprob (I2)
- numéro-analyse-sensibilité (I2)
- numéro du préordre global déterminé (I2)
- nmpcr (I2)
- le nom de cette analyse de sensibilité (A30).

}

clef d'accès

Numéro-unité : 66.

Commentaire : un article de ce fichier correspond à une analyse de sensibilité à laquelle on a appliqué la méthode hiérarchique des comparaisons par paires de Saaty.

PREORDRE-ACT-CRT-SAATY.DAT

- nmprob (I2)	}	clef d'accès
- numéro-analyse-sensibilité (I2)		
- numéro du critère (I2)		
- nmpact (I2).		

Numéro-unité : 74.

III.2.7 Les fichiers associés à la méthode des rangements médians.

METHODE-MEDIAN.DAT

- nmprob (I2)	}	clef d'accès
- numéro-analyse-sensibilité (I2)		
- type-préordre-critère (I1)		
- le nom de cette analyse de sensibilité (A30).		

Numéro-unité : 66.

Commentaire : un article de ce fichier correspond à une analyse de sensibilité à laquelle on a

appliqué la méthode des rangements médians.
 type-préordre-critère vaut : 1 si l'utilisateur a exprimé l'importance relative de chaque critère en donnant sa position sur une échelle ordinale;

2 s'il donne
 à chaque critère un poids précis.

METH-MEDIAN-PREORDRE-GLOBALE.DAT

- | | | |
|---|---|--------------|
| - nmprob (I2) | } | clef d'accès |
| - numéro-analyse-sensibilité (I2) | | |
| - numéro-ordre-critère (I2) | | |
| - numéro-ordre (I2) | | |
| - numéro du préordre global moyen (I2). | | |

Numéro-unité : 67.

Commentaire : un article de ce fichier correspond à l'existence, pour une analyse de sensibilité donnée et pour un système de poids $\{p_1, \dots, p_{k1}\}$, d'un préordre global médian.

Numéro-ordre-critère permet de distinguer les différents systèmes de poids qui peuvent être éventuellement associés à une même analyse de sensibilité.

Numéro-ordre permet de distinguer les différents préordres globaux médians, lorsque plusieurs ont été déterminés pour un même système de poids $\{p_1, \dots, p_{k1}\}$.

METH-MEDIAN-PREORDRE.DAT

- | | | |
|-----------------------------------|---|--------------|
| - nmprob (I2) | } | clef d'accès |
| - numéro-analyse-sensibilité (I2) | | |
| - numéro-ordre (I2) | | |
| - nmpact (I2) | | |
| - distance-Kemeny (F8.2). | | |

Numéro-unité : 68.

Commentaire : ce fichier permet d'enregistrer pour une analyse de sensibilité donnée et pour chaque système de poids $\{p_1, \dots, p_{k_1}\}$ qui lui est associé :

- le numéro du préordre sur les critères correspondant à ce système de poids,
- la distance de Kemeny entre les k_1 préordres sur les actions et la(les) médiane(s).

PREORDRE-CRT.MEDIANE.DAT

- | | | |
|-----------------------------------|---|--------------|
| - nmprob (I2) | } | clef d'accès |
| - numéro-analyse-sensibilité (I2) | | |
| - numéro-ordre (I2) | | |
| - numéro du critère classé (I2). | | |

Numéro-unité : 69.

Commentaire : si numéro-ordre vaut i , $1 \leq i \leq k1$,
numéro du critère classé indique le
numéro du critère qui se trouve classé
en i ème position sur l'échelle ordinale.

PREORDRE-ACT-CRT-MEDIAN.DAT

- nmprob (I2)	}	clef d'accès
- numéro-analyse-sensibilité (I2)		
- numéro-critère (I2)		
- nmpact (I2).		

Numéro-unité : 74.

III.2.8 Les fichiers associés à la méthode UTA I.

CRT-METH-UTA.DAT

- nmprob (I2)	}	clef d'accès
- numéro-analyse-sensibilité (I2)		
- numéro du critère (I2)		
- seuil de préférence si (F4.2)		
- le nombre de points α_i (I2)		
- le numéro d'ordre de la fonction d'utilité (I2).		

Numéro-unité : 74.

Commentaire : pour chaque critère i sont, en outre, en-
registrés : le numéro de la fonction d'uti-
lité qui a été déterminée, ainsi que le nom-
bre α_i de points de cette fonction qui ont
été estimés.

RAP-CRT-ANALYSE-POST-OPT11.DAT

- nmprob (I2)
 - numéro-analyse-post-optimale (I2)
 - numéro du critère (I2)
 - valeur de P_i (I1)
 - le numéro d'ordre de la fonction d'utilité (I2).
- } clef d'accès

Numéro-unité : 70.

Commentaire : pour chaque critère i associé à une analyse post-optimale, sont enregistrés : le numéro de la fonction d'utilité déterminée par cette analyse, ainsi que la valeur de P_i , coefficient de la fonction objectif (cfr. chp. III.3.3).

III.2.9 Les fichiers associés à la méthode UTA II.

RAP-CRT-ANALYSE-POST-OPT12.DAT

- nmprob (I2)
 - numéro-analyse-optimale (I2)
 - numéro du critère (I2)
 - valeur de P_i (I1)
 - valeur du poids p_i (F4.2).
- } clef d'accès

Numéro-unité : 70.

Commentaire : pour chaque critère i associé à une analyse post-optimale, sont enregistrés :

la valeur de ρ_i , coefficient de la fonction objectif (cfr. chp. III.3.3), ainsi que le poids du critère i déterminé par cette analyse post-optimale.

CRT-METHODE-UTA12.DAT

- | | | |
|-----------------------------------|---|--------------|
| - nmprob (I2) | } | clef d'accès |
| - numéro-analyse-sensibilité (I2) | | |
| - numéro du critère (I2) | | |
| - valeur du poids ρ_i (F4.2) | | |
| - nmpact (I2). | | |

Numéro-unité : 74.

III.2.10 Les fichiers communs aux méthodes UTA I et UTA II.

METHODE-UTA.DAT

- | | | |
|--|---|--------------|
| - nmprob (I2) | } | clef d'accès |
| - numéro-analyse-sensibilité (I2) | | |
| - type-méthode (I1) | | |
| - sigma (F4.2) | | |
| - numéro du préordre global (I2) | | |
| - nom de cette analyse de sensibilité (A30). | | |

Numéro-unité : 66.

Commentaire : un article de ce fichier correspond à une analyse de sensibilité à laquelle on a

appliqué la méthode UTA I si type-méthode = 1,
ou la méthode UTA II si type-méthode = 2.

RAP-UTA-ACT.DAT

- nmprob (I2)	}	clef d'accès
- numéro-analyse-sensibilité (I2)		
- type-méthode (I1)		
- numéro-ordre-action (I2)		
- l'erreur relative δ à l'action (F4.2).		

Numéro-unité : 69.

Commentaire : un article du fichier associe à chaque action,
l'erreur relative déterminée par la résolu-
tion du programme linéaire associé à une
analyse de sensibilité.

ANALYSE-POST-OPTIMALE.DAT

- nmprob (I2)	}	clef d'accès
- numéro-analyse-post-optimale (I2)		
- vaxmix (F3.0)		
- seuil (F5.3)		
- distance-Kendall (F5.2)		
- nom de l'analyse post-optimale (A30).		

Numéro-unité : 68.

Commentaire :

- vaxmix : variable qui indique si la fonction-objectif associée à une analyse post-optimale est à maximiser (vaxmix = -1) ou à minimiser (vaxmix = 1);
- distance-Kendall : valeur du τ de Kendall entre le préordre global initial et le préordre déterminé par l'analyse post-optimale;
- seuil : valeur du seuil k (F^*).

RAP-ANALY-POST-OPT-ACT.DAT

- nmprob (I2)
 - numéro-analyse-post-optimale (I2)
 - numéro-ordre-action (I2)
 - l'erreur relative δ à l'action (F4.2).
- } clef d'accès

Numéro-unité : 71.

Commentaire : un article du fichier associé à chaque action, l'erreur relative déterminée par la résolution du programme linéaire associé à une analyse post-optimale.

SURCLASSEMENT-UTA.DAT

- type-méthode (I1)
 - nmprob (I2)
 - numéro-analyse-sensibilité (I2)
 - numéro d'ordre de la relation (I2)
 - indic (I1).
- } clef d'accès

Numéro-unité : 79.

Commentaire : un article de ce fichier correspond à l'existence d'une relation de surclassement floue établie à partir de n fonctions d'utilités; indic vaut 1 si la fonction d'utilité associée à l'analyse de sensibilité est utilisée pour la construction de la relation de surclassement floue.

RAP-SURCLASSEMENT-UTA.DAT

- type-méthode (I1)	}	clef d'accès
- nmprob (I2)		
- numéro-analyse-sensibilité (I2)		
- numéro d'ordre de la relation (I2)		
- numéro-action-ligne (I2)		
- numéro-action-colonne (I2)		
- valeur du degré de crédibilité (F4.2).		

Numéro-unité : 75.

Commentaire : ce fichier enregistre le degré de crédibilité, pour chaque couple d'action (a,b), associé à la relation de surclassement floue sur les n fonctions d'utilités.

SURCLAS-LAMDA.DAT

- type-méthode (I1)
 - nmprob (I2)
 - numéro-analyse-sensibilité (I2)
 - numéro d'ordre de la relation (I2)
 - numéro d'ordre (I2)
 - valeur du seuil λ (F3.1).
- } clef d'accès

Numéro-unité : 77.

Commentaire : un article de ce fichier correspond à l'existence d'une relation de surclassement non-floue déterminée à partir d'un seuil λ , et d'une relation de surclassement floue.

RAP-SURCLUTA-ANALY-PST-OPT.DAT

- type-méthode (I1)
 - nmprob (I2)
 - numéro-analyse-sensibilité (I2)
 - numéro d'ordre de la relation (I2)
 - numéro d'ordre (I2)
 - numéro-analyse-post-optimale (I2).
- } clef d'accès

Numéro-unité : 78.

Commentaire : ce fichier permet de déterminer les analyses post-optimales pour lesquelles les fonctions d'utilités correspondantes ont été choisies pour construire une relation de surclassement floue.

RAP-UTA-ANALYSE-POST-OPT.DAT

- | | | |
|--------------------------------------|---|--------------|
| - nmprob (I2) | } | clef d'accès |
| - numéro-analyse-sensibilité (I2) | | |
| - type-méthode (I1) | | |
| - numéro d'ordre (I2) | | |
| - numéro-analyse-post-optimale (I2). | | |

Numéro-unité : 67.

Commentaire : ce fichier permet d'enregistrer, pour une analyse de sensibilité donnée, le numéro de chacune des analyses post-optimales qui a été réalisée.

III.2.11 Les fichiers associés à la méthode MAL.

Nous avons limité le nombre maximum de niveaux dans l'arborescence entre les critères à 9. Le premier niveau correspond à la racine de cette arborescence, c'est-à-dire au critère global; au dernier niveau ne sont associés que des critères élémentaires (CE).

Pour chaque noeud, le nombre de critères immédiatement subordonnés (CIS) est limité théoriquement à 99.

Nous avons défini un fichier "NIVEAU.DAT" qui permet, pour chaque critère appartenant à un niveau donné, d'indiquer s'il est élémentaire ou non.

S'il est élémentaire, le fichier "FONCTION-EFFICACITE.DAT" enregistre :

- le type de méthode utilisée pour déterminer la fonction d'efficacité associée au critère;
- le numéro du préordre entre les actions que l'utilisateur aura choisi pour le critère.

S'il n'est pas élémentaire, le fichier "NON-ELEMENTAIRE.DAT" enregistre :

- le numéro du préordre entre les critères subordonnés à ce critère;
- le numéro du préordre entre les actions qui traduit l'efficacité de chaque action pour ce critère;
- le nom de ce critère.

Les critères élémentaires sont considérés comme étant les critères du problème : leurs noms sont enregistrés, dès lors, dans le fichier "CRITERE.DAT".

Les fichiers suivants ont été définis :

METHODE-MAL.DAT

- | | | |
|---|---|--------------|
| <ul style="list-style-type: none"> - nmprob (I2) - numéro-analyse-sensibilité (I2) - la valeur de r (F8.4) - le numéro du préordre global sur les actions (I2) - le numéro du préordre sur les critères surbordonnés (I2). | } | clef d'accès |
|---|---|--------------|

Numéro-unité : 66.

Commentaire : un article de ce fichier correspond à une analyse de sensibilité à laquelle on a appliqué la méthode MAL.

NIVEAU.DAT

- nmprob (I2)	}	clef d'accès
- numéro-analyse-sensibilité (I2)		
- numéro-niveau (I1)		
- numéro-ordre-critère (I2)		
- indic (I1).		

Numéro-unité : 67.

Commentaire : indic vaut : 1 si le critère de numéro "numéro-ordre-critère", associé au niveau dont le numéro est "numéro-niveau" est élémentaire;
0 sinon.

FONCTION-EFFICACITE.DAT

- nmprob (I2)	}	clef d'accès
- numéro-analyse-sensibilité (I2)		
- numéro-niveau (I1)		
- numéro-ordre-critère (I2)		
- type-méthode (I1)		
- numéro du préordre entre les actions choisis (I2).		

Numéro-unité : 68.

Commentaire : type-méthode vaut :

- 1 si la fonction d'efficacité pour ce critère a été déterminée par la méthode des comparaisons par paires;
- 2 si la fonction d'efficacité pour ce critère a été déterminée par la méthode de construction d'une fonction d'utilité.

NON-ELEMENTAIRE.DAT

- nmprob (I2)
- numéro-analyse-sensibilité (I2)
- numéro-niveau (I1)
- numéro-ordre-critère (I2)
- numéro du préordre entre les actions (I2)
- nom du critère (A30).

}
clef d'accès

Numéro-unité : 69.
