

Table des matières

Remerciements	ix
Acronymes	xi
Notations et nomenclature	xiii
Alternatives, critères	xiii
Relations	xiv
Notations concernant la problématique de tri	xiv
Classes de modèles et paramètres	xv
Performances sur des sous-ensembles de critères	xvi
Programmes mathématiques	xvi
Note sur le (més)usage de la langue	xvii
Résumé en français	xix
Résumé en anglais	xxi
I Positionnement du travail	1
1 Positionnement du travail	3
1.1 Introduction à l'AMCD	3
1.1.1 Contexte	3
1.1.2 Alternatives, critères : définitions formelles	4
1.1.3 Utilité de l'AMCD	5
1.1.4 Classes de modèles et modèles de préférence	6
1.1.5 Classes de modèles et définition des raisonnements ac- ceptables	8
1.1.6 Préférences révélées, préférences exprimées	9
1.1.7 Le mythe du choix objectif	11
1.2 Structuration du problème de décision	11
1.2.1 Cinq problématiques de décision	12
1.2.2 Exemples de problématiques	13

Table des matières

1.2.3	Distinctions entre problématiques	14
1.2.4	Remarque concernant les termes alternatives, critères, attributs	14
1.2.5	Notations concernant la problématique de tri	15
1.3	Approche visant à obtenir un modèle précis	16
1.3.1	Incomplétude des préférences	17
1.3.2	Cas où cette approche est justifiée	20
1.4	Comparaison avec le constructivisme	20
1.4.1	L'approche constructiviste de l'aide à la décision	20
1.4.2	Deux visions compatibles	21
1.5	Approche visant à obtenir un modèle imprécis	22
1.6	Approche visant à obtenir un ensemble de modèles	23
1.7	Désagrégation des préférences	25
1.8	Traitement des inconsistances	26
1.9	L'AMCD pour la décision de groupe	27
1.9.1	Positionnement du travail	27
1.9.2	Travaux proches	28
1.9.3	Traitement des incohérences dans le cas de la décision de groupe	29
1.10	Motivation et contenu du travail	30
1.10.1	Motivation	30
1.10.2	Contenu du travail	31
2	Revue de modèles de tri et de méthodes de désagrégation	35
2.1	Classes de modèles de tri non basées sur le surclassement	36
2.1.1	Modèle de tri à fonction croissante et à seuils	36
2.1.2	Tri à base de règles	39
2.1.3	Modèle de tri à base d'intégrale de Sugeno	40
2.2	Classes de modèles proches de ÉLECTRE TRI	42
2.2.1	Concepts communs aux modèles de type ÉLECTRE TRI	42
2.2.2	Modèle ÉLECTRE TRI	45
2.2.3	Modèle MR Sort	48
2.2.4	Modèle NCSM sans veto	51
2.2.5	Modèle NCSM avec veto	54
2.3	Méthodes relatives à la problématique de tri	55
2.3.1	Désagrégation par la méthode UTADIS	55
2.3.2	Approche robuste par sous-ensemble de paramètres	57
2.3.3	Désagrégation de modèles de type ÉLECTRE TRI	60
2.3.4	Analyse d'inconsistance	61
2.3.5	Désagrégation pour des groupes	64

II	Contributions méthodologiques	67
3	Convergence de modèles MR Sort pour la décision de groupe	69
3.1	Élicitation de paramètres pour méthode MR Sort	69
3.1.1	Contexte et objectif	69
3.1.2	Un exemple de processus d'aide à la décision	72
3.1.3	ICL : Infer Category Limits	73
3.1.4	ICLV : Inferring Category Limits with Vetoes	77
3.1.5	CWR : Compute Weights Restriction	79
3.1.6	Exemple illustratif	80
3.1.7	Performance des algorithmes	84
3.1.8	Conclusions et perspectives	90
3.2	Calcul de similarité entre deux modèles MR Sort	92
3.2.1	En considérant toutes les catégories	93
3.2.2	Par paires de catégories	95
3.3	Application réelle : échelle de risque pour produits polluants	96
3.3.1	Contexte	96
3.3.2	Mode d'interrogation	97
3.3.3	Critères et catégories	98
3.3.4	Choix du type de modèle	99
3.3.5	Analyse des divergences	99
3.3.6	Modèles proposés	102
3.3.7	Affectations par ces modèles	103
3.3.8	Discussion	103
4	Crédibilités valuées et exploitation simple	105
4.1	Contexte	105
4.2	La procédure Argument Strength Assessment (ASA)	107
4.2.1	ASA ordinale	107
4.2.2	Propriétés ordinales	108
4.2.3	ASA cardinale	110
4.3	Exploitation	111
4.3.1	Illustration de l'exploitation : ensemble X	111
4.3.2	Illustration de l'exploitation : ensemble A	112
4.3.3	Description formelle	115
4.3.4	Propriété de fonctions d'affectations emboîtées	118
4.3.5	Performance des calculs avec MR Sort	118
4.4	Exemple illustratif	119
4.5	Conclusion	121

5	Une méthode de sélection de portefeuilles équilibrés	127
5.1	Introduction à la sélection de portefeuilles	127
5.2	Contexte du travail	128
5.2.1	Exemples d'applications	129
5.2.2	Travaux proches	129
5.2.3	Définition du portefeuille par une fonction de tri	130
5.3	Formulation du programme mathématique	131
5.4	Exemple illustratif	133
5.5	Conclusion et perspectives	138
5.5.1	Caractéristiques distinctives de l'approche	138
5.5.2	Une alternative à la discrimination positive	139
5.5.3	Perspectives	140
III	Implémentation	141
6	Contributions sous forme de logiciels	143
6.1	Introduction	143
6.1.1	Le format XMCDA	144
6.1.2	Les services web XMCDA	145
6.2	Bibliothèques J-MCDA et JLP	146
6.3	Composants de calcul d'une relation de surclassement	147
6.3.1	Concordance	147
6.3.2	Préférence	149
6.3.3	Discordance	149
6.3.4	Surclassement	149
6.3.5	Tri	150
6.3.6	Flux	150
6.4	Désagrégation d'un modèle MR Sort	151
6.5	Convergence de modèles pour l'évaluation de risques	151
6.5.1	Un scénario de construction d'une échelle de risque	153
6.5.2	Implémentation du scénario	156
7	Modèle de données pour l'AMCD	159
7.1	Introduction	159
7.1.1	Différents schémas pour différents usages	159
7.1.2	Un encodage moins ambigu	160
7.1.3	Réutilisation des mécanismes XML classiques	161
7.1.4	Portée du modèle de données et nomenclature	161
7.2	Modèle de données	162
7.2.1	Entités à clés	162

7.2.2	Mesures	163
7.2.3	Relations	163
7.2.4	Critères	165
7.2.5	Fonctions	166
7.2.6	Entités représentant des ensembles d'autres entités . .	169
7.2.7	Vue d'ensemble	169
7.3	Utilisation du modèle de données	169
7.3.1	Encodage des préférences et des performances	171
7.3.2	Enchaînement de composants	176
7.4	Discussion	178
8	Modélisation et résolution des programmes mathématiques	181
8.1	Erreurs numériques dans les programmes mathématiques . . .	181
8.2	Contexte	182
8.2.1	Constantes utilisées dans les programmes	183
8.3	ICL	184
8.3.1	Contraintes	184
8.3.2	Réduction de la taille du programme mathématique . .	187
8.4	ICLV	189
8.4.1	Contraintes	189
8.4.2	Vétos et critères de poids nul	191
8.4.3	Programme mathématique réduit	191
8.5	Contraintes de portefeuilles	192
	 Conclusions et perspectives	 195
9	Conclusions et perspectives	197
9.1	Conclusions	197
9.2	Conception de modèles de préférence réfléchie	199
9.2.1	Modéliser les préférences réfléchies	201
9.2.2	Deux types de préférences réfléchies	202
9.2.3	Les préférences réfléchies pourraient être incomplètes .	203
	 Bibliographie	 207

Annexes	225
A Calcul du surclassement dans d'autres modèles Électre	227
A.1 ÉLECTRE IV	227
A.2 ÉLECTRE II	228
A.3 ÉLECTRE III	228
B Brève description des modèles Prométhée I et Prométhée II	229
B.1 Calcul de la préférence	229
B.2 Exploitation	230
C Remarque concernant les vétos sur critères de poids nul	231
D Données publiées en format électronique	233

Remerciements

Je voudrais commencer ces remerciements en m'abstenant de remercier une personne qui m'est particulièrement chère. Je ne peux pas remercier Charline parce que ce qu'elle m'a apporté pendant ces trois années de thèse ne se résume pas à ses relectures ou à son aide concrète et sa patience, pourtant effectives, dans les périodes où j'ai été particulièrement pris par le travail. Je ne peux pas remercier Charline parce qu'un merci, imprimé à travers toute la page en caractères énormes, serait tellement minuscule. Je ne peux pas remercier Charline parce qu'il est absurde de remercier quelqu'un d'être ce qu'il est. Mais elle est pour beaucoup dans le plaisir que m'ont apporté ces trois dernières années. Et avec mes excuses à Montaigne, si l'on me demande avec insistance de dire pourquoi notre relation me rend si heureux, je sens que cela ne peut s'exprimer qu'en répondant : parce que c'est elle, parce que c'est moi.

Je connais Marc Pirlot depuis que je suis entré dans le monde de la recherche en 2003. Sa sagesse, sa capacité d'écoute, son intérêt pour le débat constituent pour moi un exemple, et c'est toujours un immense plaisir pour moi d'être en désaccord avec lui. La confiance qu'il m'a accordé très tôt a beaucoup compté pour moi. Si la délation était permise, je dirais qu'il est à compter comme un des principaux responsables du fait que je sévisse encore dans la recherche, et qu'il n'est de surcroît pas étranger à l'aboutissement actuel de ce méfait sous la forme de ce document de thèse.

Stéphane a largement contribué à rendre mon séjour en France très agréable et instructif. J'ai énormément bénéficié de son envie insatiable de progresser dans des débats compliqués, qu'ils soient politiques ou plus largement scientifiques. Je suis aussi très heureux d'avoir pu échanger longuement avec Alexandre, un adversaire coriace qui, suivant Socrate, m'a fait à plusieurs reprises prendre conscience que je ne savais pas alors que je croyais savoir. Il a malheureusement viré bordelais avant que nous ne venions à bout de la question des OGMs. Mais le débat continue.

J'ai eu la chance de rencontrer d'autres adversaires tout aussi coriaces, mais sur un autre plan, en particulier en les personnes de Vincent et Alicia, Laurent et Claudia, Nicolas N. Il y a peut-être lieu de démarrer une étude sociologique en la matière, car ces partenaires de jeu ont pour point commun d'être d'une gentillesse que n'égale que la férocité de Vincent lorsqu'il fond sur une victime (presque) innocente à Trône de Fer (en n'omettant pas, tout de même, de

Remerciements

signaler qu'il est désolé de devoir le trahir).

Les nombreuses soirées et moments de rires partagés avec les pré-cités et nombre d'autres amis parisiens, au rang desquels Kathleen, Xavier, Ana, Laurent P., Michel, Nico et Marc, et beaucoup d'autres, resteront pour moi des souvenirs chargés d'émotion.

Merci à Tommi pour ses techniques de développement de la créativité lors de l'écriture d'articles. La collaboration avec Tommi est toujours un plaisir, pour moi comme, je pense, pour les tenanciers de bars des environs de nos lieux de rencontre.

Je voudrais aussi remercier les nombreux collègues qui créent une excellente ambiance au LGI. Je me souviendrai en particulier de la générosité de Guillaume et Yann, toujours à l'écoute et attentifs au bien-être de leur prochain et prêts à rendre service. Le plaisir au travail n'aurait pas été le même sans la jovialité et la gentillesse de Delphine, de Carole, de ma Sylvie préférée, et de beaucoup d'autres personnes du laboratoire. Merci bien sûr à Pascal, et d'avance, bravo pour le Goncourt.

Ce travail existe grâce à Vincent, qui m'a accordé sa confiance et a œuvré à établir les conditions de sa mise en place. Mais il a fait bien plus. J'ai beaucoup appris grâce à ses conseils, ses idées, surtout grâce à sa conception de l'encadrement qui l'entraîne à consacrer du temps à ses étudiants sans les priver de leur liberté. Je souhaite à tout thésard de pouvoir bénéficier d'un encadrement aussi propice à l'épanouissement. Merci, Vincent.

Je voudrais enfin remercier les membres du jury, et Wassila pour ses nombreux apports.

Acronymes

- AMCD** Aide Multicritère à la Décision
- ASA** Argument Strength Assessment
- DM** Decision Maker
- DRSA** Dominance-based Rough Set Approach
- DSS** Decision Support Systems
- EJOR** European Journal of Operational Research
- LNCS** Lecture Notes in Computer Science
- MCDA** Multicriteria Decision Aid
- MIP** Mixed Integer Program
- NCSM** Non Compensatory Sorting Model
- PL** Programme Linéaire
- PLNE** Programme Linéaire en Nombres Entiers
- PM** Programme Mathématique
- PMML** Predictive Model Markup Language
- RESS** Reliability Engineering & System Safety
- SMAA** Stochastic Multicriteria Acceptability Analysis
- URPDM** Uncertainty and Robustness in Planning and Decision Making

Notations et nomenclature

Ce document utilise les notations et définitions suivantes. Les notations sont introduites au fur et à mesure de la lecture : cette section sert de référence uniquement.

Alternatives, critères

\mathbb{A} L'espace des alternatives, éventuellement infini, comprenant toutes les alternatives envisagées dans le problème de décision. Sauf indication contraire, nous nous intéressons à une problématique de tri visant à définir une fonction de tri sur \mathbb{A} , c'est-à-dire à définir l'affectation de n'importe quelle alternative de l'espace \mathbb{A} .

$A \subseteq \mathbb{A}$ Un ensemble d'alternatives concrètes. L'ensemble A , généralement fini, représente un ensemble d'alternatives donné dont l'affectation intéresse particulièrement les décideurs.

$X \subseteq \mathbb{A}$ Un ensemble d'alternatives utilisées en exemples dans une approche de désagrégation.

\mathcal{J} L'ensemble des critères.

$j \in \mathcal{J}$ Un critère.

$J \subseteq \mathcal{J}$ Un ensemble de critères, appelé une coalition de critères, ou simplement coalition car il s'agit du seul type de coalition auquel nous ferons référence sous ce terme.

g_j Une fonction critère $g_j : \mathbb{A} \mapsto X_j$ décrivant les performances des alternatives.

a_j La performance $g_j(a)$ de l'alternative $a \in \mathbb{A}$ sur le critère $j \in \mathcal{J}$ sera également notée a_j , lorsque cette notation ne prête pas à confusion.

X_j Le type de performances utilisées pour le critère j . Il s'agit généralement de \mathbb{R} ou d'un sous-ensemble de \mathbb{R} , auquel cas le critère est dit numérique. Il peut aussi s'agir d'étiquettes, par exemple $X_j = \{\text{mauvais, moyen, bon}\}$.

\succeq_j Une relation binaire sur $X_j \times X_j$ indiquant l'ordre de préférence sur les critères : $a \succeq_j b$ indique que l'alternative a est considérée par le décideur

Notations et nomenclature

comme étant au moins aussi bonne que b d'après le critère j . Dans ce travail, les relations \succeq_j sont supposées connues et complètes, $\forall j \in \mathcal{J}$.

$\underline{g}_j, \overline{g}_j$ Étant donné un sous-ensemble fini d'alternatives $A \subseteq \mathbb{A}$, les évaluations extrêmes sur X_j sont notées $\underline{g}_j = \min_{a \in A} g_j(a)$ et $\overline{g}_j = \max_{a \in A} g_j(a)$.

Dans le cadre de modèles de tri de type ÉLECTRE TRI, un ensemble de valeurs pour les paramètres de poids et le seuil de majorité, $\langle w_j, j \in \mathcal{J}, \lambda \rangle$, sont appelés un ensemble de coalitions de critères gagnantes, ou simplement ensemble de coalitions lorsque cela n'est pas ambigu.

Relations

Une relation binaire entre deux ensembles E_1 et E_2 est une application de $E_1 \times E_2 \mapsto \{0, 1\}$. Une relation binaire peut également être vue comme un ensemble de paires : une paire $(e_1, e_2), e_1 \in E_1, e_2 \in E_2$ appartient à la relation ssi l'application vaut 1 en cette paire. Nous utiliserons généralement des relations binaires définies sur les deux mêmes ensembles, donc application de $E \times E$, auquel cas nous parlerons simplement d'une relation binaire définie sur E si cela n'entraîne pas d'ambiguïté. Une relation binaire R définie sur un ensemble E est complète ssi $e_1 R e_2 \vee e_2 R e_1, \forall (e_1, e_2), e_1, e_2 \in E$.

Nous adopterons parfois la notation fonctionnelle : $R(e_1, e_2)$ pour désigner la valeur de la relation en le point (e_1, e_2) . Par abus de notation, nous utiliserons aussi $R(e_1, e_2)$ pour indiquer un prédicat logique lorsque cela n'entraîne pas d'ambiguïté : le prédicat $R(e_1, e_2)$ est vrai ssi l'application vaut 1 en le point (e_1, e_2) .

Une relation floue sur un ensemble E est une application de $E \times E \mapsto [0, 1]$.

Notations concernant la problématique de tri

\mathcal{C} L'ensemble de catégories auxquelles les alternatives doivent être affectées.

$C \in \mathcal{C}$ Une catégorie.

$n_{\mathcal{C}}$ Le nombre de catégories.

$\preceq, \triangleleft, \succeq, \triangleright$ Nous postulerons un ordre de préférence total sur les catégories.

Les catégories peuvent donc être numérotées selon leur position dans l'ordre de préférence (C_1, C_2, C_3 , etc.). Par convention, les catégories sont numérotées dans l'ordre de préférence croissant, C_1 désignant donc la moins bonne catégorie. Il y a $n_{\mathcal{C}}$ catégories. Pour alléger les notations, nous éviterons l'usage de l'indice associé aux catégories lorsqu'il n'est pas nécessaire. L'ordre de préférence sur les catégories est noté comme

suit : $\forall C, C' \in \mathcal{C}, C \triangleleft C'$ indique que C est une catégorie moins bonne que C' , $C \trianglelefteq C'$ indique que C est moins bonne ou égale à C' , $C \triangleright C'$ indique que C est meilleure que C' .

\mathcal{C}_n La meilleure catégorie, notée également C_{nc} lorsqu'une confusion est possible.

$[\underline{C}, \overline{C}]$ Nous utiliserons parfois des affectations en des intervalles de catégories plutôt qu'en une catégorie. Un intervalle de catégorie $[\underline{C}, \overline{C}]$ est un sous-ensemble de catégories contiguës dans \mathcal{C} :

$$[\underline{C}, \overline{C}] = \{C \in \mathcal{C} \mid \underline{C} \trianglelefteq C \trianglelefteq \overline{C}\}.$$

\mathcal{C}_I L'ensemble des intervalles sur \mathcal{C} .

I Un intervalle sera parfois noté I lorsque les catégories extrêmes n'importent pas.

f Souvent une fonction de tri précise $f : \mathbb{A} \mapsto \mathcal{C}$ ou imprécise $f : \mathbb{A} \mapsto 2^{\mathcal{C}}$. En dehors du cadre du tri, la lettre peut être utilisée pour désigner une fonction d'une autre sorte.

$a \in C$ Étant donné une fonction de tri précise f , la notation $a \in C$ désigne que l'alternative a est affectée à la catégorie C par cette fonction.

\succeq Nous manipulerons souvent au lieu d'une telle fonction de tri une relation \succeq sur l'espace des alternatives et l'ensemble des catégories : $a \succeq C$ signifie que l'alternative a est affectée au moins à la catégorie C . La relation \succeq , dans le cadre de ce travail, satisfait toujours : $a \succeq C \Rightarrow a \succeq C', \forall C' \trianglelefteq C$. Une telle relation est équivalente à une fonction de tri : $a \in C \Leftrightarrow a \succeq C \wedge [a \not\succeq C', \forall C' \triangleright C]$.

Classes de modèles et paramètres

M Une classe de modèles, donc un ensemble de fonctions de préférence. Dans le cadre du tri, il s'agit d'un ensemble de fonctions de tri. Le texte indique s'il s'agit de fonctions précises ou imprécises, si cela a une importance.

ω Un ensemble de paramètres caractérisant complètement une fonction de préférence. La définition précise de ce que contient un tel ensemble de paramètres dépend de la classe de modèles considérée. Par exemple, dans la classe des modèles de somme pondérée, ω désigne un ensemble de valeurs de poids associées chacune à un des critères $j \in \mathcal{J}$.

Ω^* L'ensemble de tous les ensembles de paramètres possibles. La définition précise de Ω^* dépend de la classe de modèles considérée. Par exemple, dans la classe des modèles de somme pondérée, Ω^* désigne tous les vecteurs de nombres entre zéro et un se sommant à un.

Notations et nomenclature

$\Omega \subseteq \Omega^*$ Un ensemble d'ensembles de paramètres.

f_ω La fonction de préférence résultant du choix des valeurs de paramètres données par ω . Dans le cadre du tri, cela peut être une fonction de tri précise ou imprécise.

f_Ω La fonction de préférence donnant des conclusions robustes considérant tous les ensembles de paramètres ω possibles dans Ω . Dans le cadre du tri, la fonction est définie comme suit : $\forall a \in \mathbb{A}$,

$$f_\Omega(a) = \bigcup_{\omega \in \Omega} f_\omega(a).$$

Performances sur des sous-ensembles de critères

Dans les sections 2.1 et 2.2, nous utiliserons les notations suivantes pour décrire les performances d'une alternative. L'ensemble $X_{-j} = \prod_{i \in \mathcal{J} \setminus \{j\}} X_i$ désigne l'espace des performances possibles en ignorant le critère j . La notation a_{-j} désigne un point de X_{-j} , donc un ensemble de performances sur tous les critères sauf j . La notation (x_j, a_{-j}) , $x_j \in X_j$, $a_{-j} \in X_{-j}$, désigne une alternative ayant la performance x_j sur le critère j et des performances correspondant à celles indiquées par a_{-j} sur les autres critères.

Programmes mathématiques

Programme Mathématique (PM) Tout programme mathématique composé de contraintes, non nécessairement linéaires, de variables continues ou entières, et d'une fonction objectif non nécessairement linéaire.

Programme Linéaire (PL) Un programme mathématique dont les contraintes sont toutes linéaires ainsi que l'éventuelle fonction objectif, les variables sont toutes continues.

Programme Linéaire en Nombres Entiers (PLNE) Un programme utilisant éventuellement des variables entières, mais constitué de contraintes s'exprimant linéairement. Cela correspond à un Mixed Integer Program (MIP) en anglais.

Note sur le (més)usage de la langue

En rédigeant ce document, j'ai adopté les rectifications orthographiques proposées par l'académie française en 1990. Par exemple, j'écris les mots *cout*, *paraître* sans accent circonflexe. Ces rectifications me semblent positives car elles visent à simplifier la langue sans l'appauvrir. Surtout, comme ces nouvelles règles sont généralement mal connues, l'adoption de cette stratégie permet d'introduire un doute dans l'esprit du lecteur qui voudrait me reprocher des fautes d'orthographe qui à n'en pas douter, malgré les relectures attentives, parsèment encore le texte. (La force de l'habitude étant tenace, je ne peux pas non plus exclure la possibilité que j'aie encore utilisé à certains endroits l'ancienne graphie.) Les modifications sont publiées sur le site de l'académie : <http://www.academie-francaise.fr/la-langue-francaise/le-francais-aujourd'hui>.

Afin de tout de même ne pas faire trop d'honneur à l'académie française, et en ma qualité de francophone belge, je me suis permis l'usage de quelques belgicisms, tels que « *endéans* », qui signifie « dans une certaine période de temps » (*endéans* la journée, *endéans* une heure), lorsque, selon mon jugement souverain, l'introduction du terme enrichit la langue (le mot « *dans* », équivalent français, est plus polysémique). À ce propos, je prie l'aimable lecteur de bien vouloir lire les nombres 73 et 95 « *septante-trois* » et « *nonante-cinq* ».

Bien que j'aie tenté d'éviter les anglicismes, je n'ai pas pu trouver de bon équivalent français pour certains termes techniques. Dans ce cas, j'ai privilégié la précision et conservé le terme anglais. Que le lecteur n'hésite pas à me faire part de ses propositions créatives, je suis intéressé.

La primauté du genre masculin en français fait beaucoup débat. L'anglais, de plus en plus souvent, marque l'indétermination du sexe d'un rôle tel que celui de décideur, très utilisé en aide multicritère à la décision, en usant alternativement de « *he* » et de « *she* », mais la pratique se transpose plus difficilement en français. En particulier, la féminisation des noms de métiers et titres pose problème dans le monde académique français (Fujimura, 2005). Du côté de l'académie française, la situation n'est pas meilleure. Voici ce que pense l'académie française de la question de la féminisation des noms de professions

et de titres (les extraits suivants sont tirés de *Le français aujourd'hui*, *ibid.*). L'académie « dénonce en particulier le contresens linguistique sur lequel repose l'entreprise : il convient de rappeler que le masculin est en français le genre non marqué et peut de ce fait désigner indifféremment les hommes et les femmes ; en revanche, le féminin est appelé plus pertinemment le genre marqué, et "la marque est privative. Elle affecte le terme marqué d'une limitation dont l'autre seul est exempt. À la différence du genre non marqué, le genre marqué, appliqué aux êtres animés, institue entre les deux sexes une ségrégation." Aussi la féminisation risque-t-elle d'aboutir à un résultat inverse de celui qu'on escomptait, et d'établir (...) une discrimination entre les hommes et les femmes. » De plus, le rapport de la commission générale de terminologie et de néologie (*ibid.*) « établit une nette différence entre les métiers d'une part (où les formes féminines sont depuis toujours en usage et ne posent pas de problème particulier), et les fonctions, grades ou titres d'autre part, qui doivent être clairement distingués de la personne. La fonction ne peut être identifiée à la personne qui l'occupe, le titre à la personne qui le porte, etc. ; pour cette raison, l'utilisation ou l'invention de formes féminines n'est pas souhaitable. »

Ces arguments me semblent peu convaincants. Le genre masculin est bien souvent réservé, pour des raisons historiques, aux fonctions et titres les plus prestigieux. Il paraît crédible que l'utilisation systématique du masculin pour parler de certaines fonctions nuise, au moins un peu, à l'accession des femmes à ces fonctions. N'étant pas expert de ce domaine, je ne puis citer d'études appuyant cette position, si tant est qu'il en existe. Mais l'académie française n'en cite pas plus. En suivant mon intuition, et en considérant que le bénéfice du doute devrait profiter aux femmes, souvent minoritaires dans les professions socialement élevées, je considère donc cette hypothèse comme raisonnable.

En guise de compromis, je parlerai du décideur au masculin, mais je me permettrai, sans autres avertissements, de supposer de temps à autres que l'utilisateur de mon programme est une utilisatrice, ou pire, que le responsable politique utilisant nos méthodes est une responsable politique, et je parlerai systématiquement de l'analyste au féminin. Ce choix me permettra de me faire des ennemis de tous les bords, puisque les lecteurs conservateurs seront outrés, tandis que les féministes me traiteront avec le plus grand mépris pour avoir adopté une position aussi molle. C'est, en somme, un compromis à la belge.

Résumé en français

Le champ de l'Aide Multicritère à la Décision (AMCD) propose des manières de modéliser formellement les préférences d'un décideur de façon à lui apporter des éclaircissements. Le champ s'intéresse aux problèmes impliquant une décision et faisant intervenir plusieurs points de vue pour évaluer les options (ou alternatives) disponibles.

Ce travail vise principalement à proposer des méthodes d'élicitation, donc des façons de questionner un décideur ou un groupe de décideurs pour obtenir un ou plusieurs modèles de préférence. Ces méthodes utilisent des techniques dites de désagrégation consistant à prendre des exemples de décision pour base de la modélisation. Dans le contexte étudié, les modèles de préférence sont des modèles de tri : ils déterminent une façon d'affecter des alternatives à des catégories ordonnées par préférence. Nous nous intéressons à la classe de modèles de tri MR Sort.

Nous présentons une méthode permettant de faire converger un groupe de décideurs vers un modèle de tri unique. Elle s'appuie sur des programmes mathématiques. Nous analysons également en détail les difficultés liées aux imprécisions numériques posées par l'implémentation de ces programmes. Nous proposons aussi un algorithme permettant de comparer deux modèles MR Sort.

Nous introduisons une manière novatrice d'interroger le décideur d'une façon qui permet de prendre en compte ses hésitations, via l'expression de degrés de crédibilités, lorsqu'il fournit des exemples d'affectation. Les résultats de la méthode permettent au décideur de visualiser les compromis possibles entre la crédibilité et la précision des conclusions obtenues.

Nous proposons une méthode de choix de portefeuille. Elle intègre des préoccupations d'évaluation absolue, afin de s'assurer de la qualité intrinsèque des alternatives sélectionnées, et des préoccupations portant sur l'équilibre du portefeuille résultant. Nous expliquons également en quoi cette méthode peut constituer une alternative à la discrimination positive.

Nous décrivons les composants logiciels réutilisables que nous avons soumis à une plateforme de services web, ainsi que les fonctionnalités développées dans une bibliothèque qui implémente les méthodes proposées dans ce travail.

Un schéma de données existe visant à standardiser l'encodage de données de méthodes d'AMCD en vue de faciliter la communication entre composants logiciels. Nous proposons une nouvelle approche visant à résoudre un certain

Résumé en français

nombre d'inconvénients de l'approche actuelle.

Nous développons en guise de perspective une proposition visant à inscrire la modélisation des préférences dans une épistémologie de type réaliste.

Résumé en anglais

The field of Multicriteria Decision Aid (MCDA) aims to model in a formal way the preferences of a Decision Maker (DM) in order to bring informations that can help her in a decision problem. MCDA is interested in situations where the available options (called alternatives) are evaluated on multiple points of view.

This work suggests elicitation methods: ways of questioning a DM or a group of DMs in order to obtain one or several preference models. These methods rely on so-called disaggregation techniques, which use exemplary decisions as a basis for building the preference model. In our context, the preference models are sorting models: they determine a way of assigning alternatives to preference-ordered categories. We are interested in a class of sorting models called MR Sort.

We present a method that helps a group of DMs converge to a unique sorting model. It uses mathematical programs. We also analyze in detail the difficulties due to numerical imprecision when implementing these programs, and we propose an algorithm allowing to compare two MR Sort models.

We introduce a novel way of interrogating the DM in order to take her hesitations into account, through the expression of degrees of credibility, when she gives assignment examples. Results of the method let the DM examine possible compromises between credibility and precision of the conclusions.

We propose a method to choose portfolios. It encompasses two dimensions: absolute evaluation, in order to ensure that the selected alternatives are sufficiently good, and balance of the resulting portfolio. We also explain how this method compares to affirmative action.

We describe the reusable software components that we have submitted to a web services platform, as well as functionalities developed in a library that implements the methods this work proposes.

A data scheme exists that aims to standardize encoding of data related to MCDA methods, in order to ease communication between software components. We propose a new approach aiming to solve some drawbacks of the current approach.

We develop as a perspective a proposal that aims to integrate preference modeling into the framework of realistic epistemology.

Première partie

Positionnement du travail

Nos sociétés s'approchent de la situation, tant rêvée depuis des siècles par les utopistes, où la malédiction du travail sera enfin écartée. Autrefois on appelait cela « l'âge d'or ». Maintenant que cet âge d'or apparaît non plus comme inaccessible, mais comme une réalité présente à l'horizon, absurde-ment on l'appelle « crise » et l'on se lamente.

(Albert Jacquard, Cinq milliards d'hommes dans un vaisseau)

1 Positionnement du travail

Ce chapitre présente une vue d'ensemble du domaine de l'Aide Multicritère à la Décision (AMCD) et indique comment ce travail se situe par rapport à la vaste littérature concernant ce champ. L'essentiel du chapitre suppose un contexte impliquant un décideur unique, par simplicité. Nous passons seulement dans un deuxième temps à l'exposé d'approches impliquant un groupe de décideurs, cas auquel ce travail s'intéresse également.

1.1 Introduction à l'AMCD

1.1.1 Contexte

Le champ de l'AMCD propose des manières de modéliser formellement les préférences d'un décideur de façon à l'éclairer dans des problèmes impliquant une décision. Les situations de décisions auxquelles nous nous intéressons ici font intervenir plusieurs critères, points de vue utilisés pour évaluer les options disponibles.

Supposons par exemple que notre décideur, s'étant récemment procuré un surcroît important de temps libre (une tendance très marquée en ces temps de crise), s'interroge sur l'utilisation qu'il pourrait faire d'un coin de jardin vacant. Il pourrait par exemple y planter des carottes, ou des tomates, etc., ou ne rien y faire du tout. Ces différentes options sont nommées des alternatives : planter des carottes est une alternative, ne rien faire en est une autre. (Il s'agit d'un abus de langage, nous y reviendrons.) Un critère intervenant dans le choix est la pénibilité du travail, un autre est la rentabilité financière, etc. Les critères intervenant dans les situations de décision auxquelles nous nous intéressons sont conflictuels : ils ne déterminent pas tous le même ordre de préférence sur les alternatives. Pour choisir, le décideur doit déterminer comment il pondère les différents critères, ou plus généralement doit trouver une méthode pour agréger les différentes dimensions considérées. Cela introduit un aspect subjectif dans le problème : deux décideurs faisant face à une même situation ne choisiront peut-être pas une même destination à leur lopin de terre, et ce même s'ils sont d'accord sur les évaluations de chaque alternative sur chaque critère.

1 Positionnement du travail

Nous appellerons performances les évaluations des alternatives sur chaque critère. La performance de l'alternative « tomates » sur le critère « rentabilité financière » est peut-être « bonne » alors que la performance de l'alternative « carottes » est peut-être « mauvaise ». Nous considérerons les performances comme données, ou objectives, c'est-à-dire ne dépendant pas des préférences du sujet. Le problème de décision fait donc intervenir des données objectives : ensemble d'alternatives considéré, critères considérés, performances ; et des données subjectives, représentant les préférences du décideur. Notons que nous n'aborderons pas dans ce travail les questions de décision dans l'incertain, ou dans le risque. Les conséquences de chaque choix, représentées par les performances sur les critères, sont supposées connues précisément. Nous ne parlons pas non plus du problème des données manquantes, les performances sont supposées connues entièrement.

1.1.2 Alternatives, critères : définitions formelles

Présentons ces concepts plus formellement. Nous supposons défini l'espace des alternatives \mathbb{A} , généralement infini, comprenant toutes les alternatives envisagées dans le problème de décision. Sauf indication contraire, nous nous intéressons à une problématique de tri visant à définir une fonction de tri sur \mathbb{A} , c'est-à-dire de définir l'affectation de n'importe quelle alternative de l'espace \mathbb{A} .

Est également défini, l'ensemble de critères \mathcal{J} . Un critère $j \in \mathcal{J}$ est associé à une fonction critère $g_j : \mathbb{A} \mapsto X_j$ décrivant les performances des alternatives. La performance $g_j(a)$ de l'alternative $a \in \mathbb{A}$ sur le critère $j \in \mathcal{J}$ sera également notée a_j , lorsque cette notation ne prête pas à confusion. L'ensemble X_j indique le type de performances utilisées pour le critère j . Il s'agit généralement de \mathbb{R} ou d'un sous-ensemble de \mathbb{R} , mais il peut aussi s'agir d'étiquettes, par exemple $X_j = \{\text{mauvais, moyen, bon}\}$.

L'espace des alternatives peut également être décrit par $\prod_{j \in \mathcal{J}} X_j$: un vecteur dans cet espace décrit l'ensemble des performances d'une alternative. Parfois, nous assimilerons donc \mathbb{A} à ce produit. Cependant, ce procédé est restrictif : l'ensemble des alternatives considéré par le décideur pourrait comprendre deux alternatives ayant les mêmes performances sur l'ensemble de critères considéré, mais que le décideur considère de façon distincte. Sauf indication contraire, l'ensemble \mathbb{A} sera donc considéré comme un ensemble d'indices d'alternatives, distinct de $\prod_{j \in \mathcal{J}} X_j$.

Nous considérerons connu l'ordre de préférence sur chaque critère, noté \succeq_j . Il s'agit d'une relation binaire définie sur X_j , où $a \succeq_j b$ indique que l'alternative a est considérée par le décideur comme étant au moins aussi bonne que b d'après le critère j . Les relations \succeq_j sont supposées complètes, $\forall j \in \mathcal{J}$.

1.1.3 Utilité de l'AMCD

Un problème de décision peut être suffisamment complexe, ou avoir des enjeux suffisamment importants, pour que le décideur doute de la décision à prendre. L'aide multicritère à la décision propose de fournir des recommandations à un décideur (Keeney et Raiffa, 1976). L'objectif de la méthode est d'aboutir à une façon de comparer qui convienne au décideur, dans le sens où elle lui donne des arguments qu'il trouve suffisamment convaincants pour valider les recommandations fournies par la méthode. Il ne s'agit pas nécessairement d'une réponse directe au problème de décision initial (un choix, dans notre exemple de jardinage), pour des raisons qui seront expliquées plus loin (Roy, 1985). Par exemple, une méthode d'AMCD peut indiquer quelles sont les informations subjectives qui amènent nécessairement à une conclusion donnée, ou indiquer si une conclusion donnée est compatible avec un mode de raisonnement donné (Figueira *et al.*, 2005a).

Dit généralement, une méthode d'AMCD a pour objectif d'apporter des éléments d'information permettant d'éclairer un problème de décision. Plus précisément, on peut noter deux formes particulières que peut prendre cet objectif. Premièrement, la modélisation formelle peut permettre au décideur de remettre ses propres préférences en question. En réfléchissant aux propriétés qu'il souhaite que ses décisions respectent, il peut se rendre compte que sa façon naturelle d'agir n'est pas en adéquation avec ces exigences, et décider éventuellement de modifier ses choix. Comme le fait remarquer Dennett (2002, 2012), une propriété remarquable du raisonnement humain est la faculté de pouvoir choisir de préférer autre chose, en somme, d'avoir des préférences sur ses propres préférences. Deuxièmement, le modèle de préférence peut contribuer à rendre une procédure de décision transparente et impartiale (le mot impartial doit être compris dans un sens différent du mot objectif, nous y reviendrons).

Considérons un terrain plus réaliste d'application des méthodes d'AMCD (bien que moins drôle que les problèmes de jardinage). Imaginons que le décideur soit responsable de la sélection des étudiants autorisés à intégrer une école donnée. Le problème comporte une dimension subjective, car il faut pondérer les différentes notes sur les différents cours, les résultats aux entretiens de motivation, les connaissances linguistiques de l'étudiant, etc. Quelle que soit la façon dont le décideur agrège les différentes dimensions considérées, le décideur souhaite probablement que cette agrégation respecte la propriété suivante, dite de dominance : si un étudiant a est moins bon qu'un étudiant b sur tous les critères considérés, il ne doit pas être considéré dans le résultat final comme meilleur que b . Pourtant, il est possible qu'une sélection effectuée par le décideur sans l'aide d'une méthode formelle ne respecte pas la dominance.

1 Positionnement du travail

Une telle situation peut arriver par exemple si le choix du décideur fait intervenir inconsciemment des caractéristiques physiques ou socio-économiques des étudiants dont il ne souhaitait pas tenir compte. Dans un exemple moins caricatural, le décideur pourrait vouloir s'assurer que la sélection n'accorde pas « trop de poids » à certains critères.

Les méthodes d'AMCD permettent en général de déterminer un modèle de préférence, qui permet de déterminer des recommandations (Bell *et al.*, 1988 ; Bouyssou *et al.*, 2006a). Dans un problème de choix tel que le problème de jardinage exposé ci-dessus, le modèle de préférence détermine une fonction qui, étant donné un ensemble d'alternatives et leurs performances, recommande une alternative comme étant la meilleure (ou plus généralement, un sous-ensemble d'alternatives comme étant potentiellement les meilleures, nous y reviendrons). Un modèle de préférence est associé à un ensemble de valeurs de paramètres utilisés pour représenter les préférences du décideur. L'obtention d'un modèle de préférence permet aussi, s'il est utilisé d'une certaine manière, de rendre une décision impartiale. Par exemple, dans le problème de sélection d'étudiants mentionné, le décideur pourrait décider de publier le modèle de préférence qui sera utilisé avant même de recevoir les dossiers des étudiants. Cela rend le processus de sélection déterministe, reproductible, et prévisible.

1.1.4 Classes de modèles et modèles de préférence

Il est commode de définir a priori une classe de modèles envisageable comme candidats à la représentation des préférences du décideur. Ceci peut être vu formellement comme une classe de fonctions associée à un ensemble de paramètres dont les valeurs déterminent le modèle de préférence (Bouyssou *et al.*, 2006a). Choisir une bonne classe de modèles est un des problèmes difficiles auxquels l'analyste en AMCD fait face (Belton et Stewart, 2002).

Considérons un problème consistant à choisir la meilleure alternative, tel que le problème de jardinage exposé. Supposons que les performances s'expriment toutes numériquement. Dans ce cas, il est possible d'utiliser par exemple la somme pondérée comme classe de modèles. Notons M cette classe de modèles. Pour choisir la meilleure alternative, une valeur $f(a)$ sera attribuée à chaque alternative $a \in \mathbb{A}$. La valeur d'une alternative $f(a)$ est la somme de ses performances sur chaque critère multipliée par le poids associé au critère, noté w_j : $f(a) = \sum_{j \in \mathcal{J}} w_j a_j$. L'alternative de valeur maximale sera retenue comme la meilleure. Nous pouvons appeler f une fonction de choix, étant donné que la connaissance de cette fonction détermine le choix de la meilleure alternative.

Dans cet exemple, les paramètres sont les poids à associer à chaque critère. Fixer des valeurs pour les poids suffit à déterminer une fonction de choix. Notons $\omega = \langle w_j, j \in \mathcal{J} \rangle$ un ensemble de valeurs de poids, et notons f_ω la fonction

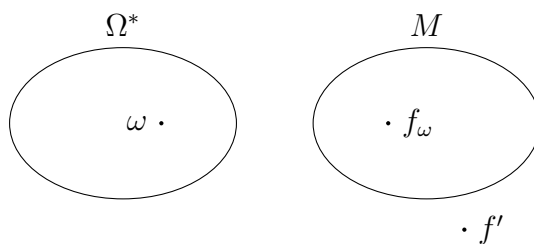


FIGURE 1.1 – Une classe de modèles, l'ensemble de paramètres associé, et deux fonctions de préférence.

calculant la valeur de chaque alternative en utilisant les poids donnés par ω . Notons Ω^* l'ensemble des paramétrages possibles pour la classe de modèle M , donc ici l'ensemble des vecteurs poids possibles. La classe de modèles de somme pondérée M contient toutes les fonctions de choix possibles fonctionnant selon les principes de la somme pondérée : $M = \{f_\omega, \omega \in \Omega^*\}$.

De manière plus générale, le problème ne consiste pas nécessairement à choisir, et la classe de modèles n'est bien sûr pas toujours la classe des modèles de somme pondérée. Nous parlerons en général de fonction de préférence f et de classe de modèles M . Dans la suite de ce travail, nous désignerons parfois simplement par le terme modèle une classe de modèles lorsque le contexte rend le terme clair. Nous pourrions par exemple parler du modèle de la somme pondérée pour désigner la classe de modèles de somme pondérée.

Dans un contexte où la classe de modèles M est connue, nous appellerons modèle de préférence le couple (ω, f_ω) : connaissant la classe de modèles M , la donnée ω capture la subjectivité du décideur et détermine une fonction f_ω représentant les préférences du décideur. La seule donnée ω pourra également être appelée modèle de préférence, étant donné que sa connaissance suffit, ayant M , à déterminer la fonction f_ω .

La figure 1.1 illustre deux cas possibles concernant la représentation des préférences d'un décideur. La figure montre une classe de modèles M , ensemble de fonctions candidates à la représentation de ses préférences. Cet ensemble peut être associé à l'ensemble Ω^* , ensemble des ensembles de valeurs que peuvent prendre les paramètres utilisés pour la classe de modèles choisie. Considérons un modèle de préférence $\omega \in \Omega^*$. Il représente un ensemble de valeurs de paramètres, et peut être associé à une fonction f_ω dans l'espace des fonctions de préférences. La fonction f_ω peut être celle qui représente les préférences du décideur. Dans un cas moins favorable, il se pourrait que la fonction représentant les préférences du décideur, f' , ne soit pas comprise dans l'ensemble M . Cela reviendrait à dire qu'il n'existe pas de valeurs de paramètres $\omega \in \Omega^*$ représentant les préférences du décideur. Voyons comment une telle situation

peut s'interpréter.

1.1.5 Classes de modèles et définition des raisonnements acceptables

Il faut noter que la classe de modèles de somme pondérée M ne contient pas toutes les fonctions de choix possibles. Par exemple, les fonctions ne respectant pas la dominance sont exclues : il est impossible de trouver un ensemble de poids qui entraîne qu'une alternative moins bonne qu'une autre sur tous les critères soit choisie.

Il est certainement raisonnable d'exclure les fonctions ne respectant pas la dominance de l'ensemble des fonctions a priori candidates à la représentation des préférences du décideur. Mais d'autres fonctions sont exclues également. Imaginons qu'on souhaite déterminer le meilleur étudiant d'un groupe en fonction de leurs performances sur deux cours notés sur vingt, et soient trois étudiants ayant les notes suivantes : (14, 14), (20, 9), (9, 20). Il est possible que le décideur préfère le premier étudiant aux deux autres : dans beaucoup de systèmes de notation scolaires, une note inférieure à 10/20 est considérée comme un échec. Or, une telle fonction de choix n'existe pas dans la classe des modèles de somme pondérée : il n'existe pas de valeurs de paramètres ω , donc de valeurs de poids, qui soient favorables au premier étudiant. En effet, préférer le premier étudiant au deuxième requiert un poids plus important sur le deuxième critère ; préférer le premier étudiant au troisième requiert un poids plus important sur le premier critère.

Déterminer une classe de modèles M , c'est également déterminer ce qui est considéré comme un mode de raisonnement a priori acceptable (à savoir toutes les fonctions dans M , et seulement celles-là). Inversément, la notion de ce qui est raisonnable, une fois bien définie, détermine l'ensemble de fonctions acceptables, c'est-à-dire les façons a priori acceptables d'agrèger les points de vue représentés par les critères pour parvenir à des conclusions. Le champ de l'AMCD propose plusieurs façons de définir ce qui est raisonnable, donc plusieurs classes de modèles. Nous verrons quelques classes de modèles courants. Comme l'AMCD a pour objectif d'aider un décideur donné, il semble judicieux de laisser le décideur maître de déterminer la bonne définition de ce qui lui semble raisonnable dans sa situation de décision. Mais ce n'est pas toujours simple, car les définitions formelles peuvent être complexes, si bien qu'on peut se demander comment interroger convenablement le décideur sur son choix de définition. De plus, certains chercheurs semblent apprécier l'idée d'imposer leur définition de ce qui est raisonnable à toute situation de décision. Cela peut susciter des débats animés entre partisans de différentes classes de mo-

dèles (Schärliig, 1996 ; Zionts, 1997 ; Roy et Vincke, 1998 ; Marchant et Pirlot, 1999). Il faut toutefois noter que les discussions ne concernent généralement pas seulement la détermination de la meilleure classe de modèles. Comme nous le verrons plus loin, différentes approches de l'AMCD sont concevables, et ceci vient compliquer les discussions concernant les classes de modèles.

Certaines classes de modèles ont été axiomatisées (Bouyssou *et al.*, 2006a). Dans notre cadre, un axiome est une condition, définie formellement, concernant l'objet modélisé, à savoir les préférences du décideur. Les conditions nécessaires expriment un ensemble de conditions que les préférences doivent satisfaire pour qu'il existe un modèle de la classe de modèles considérée qui puisse les représenter, c'est-à-dire, pour qu'il existe une fonction dans la classe de modèles qui représente ces préférences. Par exemple, respecter la dominance est une condition nécessaire pour que la classe des modèles de somme pondérée puisse représenter les préférences du décideur. Il en va de manière similaire avec les conditions suffisantes : si la préférence du décideur satisfait ces conditions, cela suffit à garantir que la classe des modèles considérée peut représenter ses préférences. Pour certaines classes de modèles, des conditions ont été trouvées qui sont à la fois nécessaires et suffisantes, et sont de surcroît facilement interprétables. Un tel système d'axiomes peut alors fournir un argument fort en faveur ou en défaveur de l'utilisation de la classe de modèles ainsi axiomatisée, selon que le décideur souhaite respecter ces conditions, ou au contraire ne les accepte pas. Les décompositions en axiomes permettent également de découvrir des relations parfois surprenantes d'inclusion entre classes de modèles, ou même l'équivalence entre deux classes de modèles apparemment distinctes. Nous en verrons des exemples au chapitre 2.

Une fois une classe de modèles définie, il reste donc à définir les valeurs des paramètres déterminant le modèle de préférence du décideur. Une des questions auxquelles tente de répondre le champ de l'AMCD est de déterminer des manières simples de paramétrer le modèle, par exemple en déterminant quelle série de questions l'analyste devrait poser au décideur. Le processus visant à obtenir des informations de préférence du décideur en le soumettant à des stimuli est appelé l'élicitation (Fischhoff, 1991). C'est un sujet important de ce travail.

1.1.6 Préférences révélées, préférences exprimées

L'AMCD s'attache à représenter les préférences du décideur dans un sens précis : il s'agit de ses préférences après réflexion, après avoir pesé le pour et le contre. Ces préférences peuvent se démarquer fortement de ses préférences spontanées. L'interaction avec l'analyste durant le processus de modélisation et les questions posées peuvent conduire le décideur à prendre des décisions

1 Positionnement du travail

qu'il n'aurait pas prises sans l'intervention de ce processus.

L'approche proposée par l'AMCD se distingue donc assez fortement, généralement, de l'approche appelée descriptive. L'approche descriptive, par exemple utilisée dans le champ de l'Apprentissage des Préférences (*Preference Learning*), vise à modéliser le comportement naturel du décideur (Fürnkranz et Hüllermeier, 2011). Par comportement naturel, il faut ici comprendre que les données de préférences utilisées, et que le modèle vise à prédire, sont issues de situations de décision réelles et souvent récoltées sur le terrain. Un exemple typique d'une telle donnée est une décision d'achat. Les économistes, en particulier, sont friands de ce genre de données qu'ils nomment informations de préférence révélées. Au contraire, les informations de préférence obtenues sous forme de réponse à des questions posées par l'analyste, utilisées habituellement en AMCD, peuvent être considérées comme non fiables par les économistes (Abdellaoui *et al.*, 2007).

Il y a donc deux manières de voir distinctes. En AMCD, l'interaction avec l'analyste, requise pour récolter les préférences, est considérée comme bénéfique puisque octroyant au décideur une opportunité de réfléchir à son problème et éventuellement de modifier sa décision. Dans une approche descriptive au contraire, il s'agit de recueillir des décisions aussi proches que possible de décisions de terrain : une modification de la décision à cause du contexte expérimental visant à analyser les préférences est considérée comme un bruit.

Notons que comme dans tout champ scientifique, les frontières des disciplines sont poreuses : nous ne prétendons pas exclure du champ de l'AMCD toute attitude descriptive, ni circonscrire l'Apprentissage des Préférences aux approches descriptives.

L'objectif de l'AMCD est donc bien distinct de l'objectif descriptif considérant uniquement les préférences révélées. Est-ce également un objectif valable ? Au risque de ne pas surprendre le lecteur : nous le pensons. Comme le font remarquer Beshears *et al.* (2008), il est dans certaines circonstances difficile de considérer les préférences révélées comme réellement révélatrices de ce que le décideur préfère. L'article cite, entre autres exemples, le cas des paiements des amendes de retard sur les cartes de crédit. Les nouveaux possesseurs d'une carte de crédit analysés par une étude citée ont une tendance à payer des amendes de retard qui s'estompe avec l'expérience grandissante. Il est vraisemblable que cela ne montre pas une préférence initiale de ces possesseurs en faveur d'un délai de paiement au prix de l'amende, mais plutôt un apprentissage.

Un autre argument donné dans cet article est l'observation expérimentale selon laquelle les sujets ont une plus forte tendance à choisir un investissement simple (sans risque) lorsque le nombre d'options d'investissement à leur disposition augmente. Il est vraisemblable que les individus ne souhaitent pas

réellement adopter une attitude plus ou moins prudente en fonction du nombre d'options auxquelles ils ont accès, mais bien plutôt, comme suggèrent les auteurs, que les individus sélectionnent plus facilement les options simples lorsque la situation devient trop complexe, de façon à échapper à cette complexité du choix.

1.1.7 Le mythe du choix objectif

Mentionnons, pour clore cette introduction, que l'AMCD est malheureusement parfois utilisée d'une façon difficilement justifiable pour donner un aspect objectif à des décisions en en masquant d'une part la dimension subjective et d'autre part l'éventuel aspect arbitraire. D'une part, les préférences contenues dans le modèle qui sert de base aux recommandations de l'analyste sont les préférences d'un décideur donné. Les méthodes analysées dans ce travail, et généralement en aide multicritère à la décision, ont justement pour caractère distinctif par rapport à d'autres sciences qu'elles incorporent la subjectivité de certains individus. Il n'est aucunement légitime de faire de ces recommandations des normes à suivre sous peine d'être « irrationnel ». Même si un individu accepte le choix de la classe de modèles représentant la rationalité (ce qu'il n'est aucunement obligé de faire, puisqu'on voit mal pourquoi l'analyste ou la scientifique qui étudie les méthodes est seule légitime pour décréter ce qui est rationnel et l'imposer à autrui), il n'est pas encore dit qu'il partage les valeurs des paramètres du modèle avec le décideur interrogé, ce qui est nécessaire pour en partager les conclusions. D'autre part, l'analyste effectue souvent elle-même des choix, car le modèle de préférence est généralement sous-déterminé par les indications des décideurs (ceci est analysé plus en détail ci-dessous). Dans ce cas, en plus des précautions indiquées précédemment, les recommandations doivent être comprises comme une des issues possibles du processus d'aide multicritère à la décision.

1.2 Structuration du problème de décision

Cette section définit certains éléments de la structuration du problème de décision et de l'approche d'AMCD dont nous aurons besoin. Sont décrites entre autres les problématiques, représentant le type d'approche choisie pour éclairer le problème de décision.

Nous supposons dans ce travail que le problème de décision qu'il s'agit d'éclairer est déjà largement structuré : l'ensemble des critères à prendre en compte a été défini, l'espace des alternatives est délimité, les performances sont connues. Structurer un problème de décision d'une telle manière est une

1 *Positionnement du travail*

tâche très complexe, que nous laissons de côté ici. De même, nous supposons que la problématique est choisie. En pratique, le processus d'aide à la décision peut consister en une succession de phases dont certaines consistent à structurer ou re-structurer le problème au vu des résultats des phases précédentes. L'ensemble des alternatives considérées, par exemple, peut être revu lorsque les critères se précisent, ou lorsque le décideur se rend compte qu'aucune des alternatives envisagées initialement n'est satisfaisante. Les outils que nous proposons peuvent ne s'appliquer qu'à une des phases du processus d'AMCD : nous ne prétendons bien sûr pas qu'ils suffisent à résoudre tout problème de décision à eux seuls.

1.2.1 **Cinq problématiques de décision**

En aide multicritère à la décision, quatre problématiques sont habituellement distinguées (Roy, 1985, 2005a) : la problématique de choix, de tri, de rangement, de description. Nous leur ajoutons la problématique de sélection de portefeuille, à laquelle ce travail s'intéresse également. Le mot problématique doit être compris comme désignant le type d'éclairage que la procédure d'aide à la décision vise à apporter au décideur.

La problématique de choix consiste à orienter l'aide vers la sélection d'un nombre aussi faible que possible de bonnes alternatives, dans un cadre où le décideur souhaite choisir une et une seule de ces alternatives. La procédure peut viser l'élimination des alternatives qui ne constituent certainement pas de bonnes candidates. Cela n'implique pas nécessairement que celles qui restent sont également valables : l'ensemble sélectionné peut contenir des alternatives qu'on ne prétend pas, ou pas encore à cette étape du processus d'aide à la décision, pouvoir comparer.

La problématique de tri oriente l'aide vers une affectation de chaque alternative à une catégorie, jugée la plus appropriée, parmi une famille de catégories prédéfinies. La catégorie à laquelle une alternative est affectée peut représenter un jugement sur la qualité de cette alternative, ou déterminer une action à effectuer (implémenter l'action représentée par l'alternative, rejeter l'alternative, l'étudier plus en détail...). Les catégories ne sont pas nécessairement ordonnées par préférence, dans la définition originale de cette problématique. Cependant, dans ce travail nous ne considérerons que le cas de catégories ordonnées. Nous élargissons la définition et envisageons également qu'une alternative puisse être affectée à un sous-ensemble de catégories, indiquant une hésitation concernant la catégorie la plus appropriée.

La problématique de rangement oriente l'aide vers l'obtention d'un rangement complet ou partiel de l'ensemble des alternatives, de la meilleure à la moins bonne.

La problématique de sélection de portefeuilles vise à proposer un ou plusieurs sous-ensembles d'alternatives cohérents, appelés portefeuilles (Salo *et al.*, 2011). Une alternative représente en général une action dont il faut choisir s'il faut l'implémenter, et un portefeuille représente en général des actions qui devraient être implémentées ensemble car elles satisfont ensemble des propriétés de cohérence. Dans ce type de problème, les alternatives ne sont pas évaluées indépendamment l'une de l'autre. Par exemple, une action peut être adéquate seulement si elle est implémentée avec une autre qui est pré-requise pour rendre la première réalisable, ou seulement si ses faiblesses sur certains critères sont contrebalancées par une autre action du portefeuille retenu. Le chapitre 5 aborde cette problématique plus avant.

La problématique de description consiste à déterminer les performances d'un ensemble d'alternatives jugées réalisables étant donné un ensemble de critères jugés adéquats pour aborder le problème de décision. Il faut généralement s'intéresser à cette problématique avant d'aborder d'autres questions, mais elle peut consister une fin en soi si les résultats de cette investigation apportent suffisamment d'informations au décideur. Nous supposons dans ce travail que cette étape est réalisée.

La formulation de ces problématiques pourrait paraître trop peu ambitieuse. Par exemple, pourquoi souhaiterait-on dans la problématique de rangement n'obtenir qu'un rangement partiel de l'ensemble des alternatives? La section 1.3 explique pourquoi l'exigence de complétude peut parfois être jugée déraisonnable. Donnons d'abord des exemples d'applications de ces problématiques.

1.2.2 Exemples de problématiques

Un gouvernement peut adopter une problématique de choix en vue d'établir une procédure de sélection d'un type de revêtement routier. Chaque type de revêtement peut être considéré comme une alternative et sera évalué sur des critères tels que son insonorisation, sa résistance, son coût, etc.

La problématique de tri peut venir en aide à un comité qui souhaiterait établir une procédure d'étiquetage écologique de produits de grande distribution tels que des frigos. En fonction de la consommation électrique, de la proportion de matériaux recyclables ou d'autres éléments, les frigos peuvent être répartis en des catégories A+, A, B, etc. associées par exemple à des niveaux de taxes croissantes.

Un comité peut souhaiter disposer d'une aide formelle afin d'établir les règles de rangement des participants à une compétition sportive. Les performances peuvent être mesurées de multiples façons, en particulier si la compétition implique plusieurs épreuves, comme dans le cas du décathlon. Établir un classement final à partir du temps passé à effectuer chaque épreuve, du nombre

1 *Positionnement du travail*

d'obstacles franchis, etc., constitue alors un problème d'agrégation pour lequel la problématique du rangement peut être utile.

Supposons qu'un groupe d'investigateurs doive choisir un ensemble de sites à visiter dans un pays I , avec pour objectif de découvrir si le pays produit des armes de destruction massive. Il ne suffit pas de choisir les sites qui, en les jugeant indépendamment, présentent le plus de chances d'héberger ce genre de programmes, mais il faut choisir un ensemble de sites qui puissent être visités étant donné les contraintes de ressources disponibles (en temps, en argent), qui couvrent bien le pays géographiquement, qui soient bien répartis en termes de types de recherches menés sur ces sites, etc. Une telle situation pourrait justifier un recours à une problématique de sélection de portefeuilles.

1.2.3 Distinctions entre problématiques

Insistons sur les distinctions entre ces problématiques. La problématique de rangement se base sur des jugements relatifs : les alternatives sont comparées entre elles. Le rangement dit quelle alternative est meilleure que telle autre, mais ne dit pas si les alternatives sont toutes bonnes, ou toutes mauvaises. Au contraire, la problématique de tri compare à des normes représentées par les catégories, et utilise donc des jugements absolus. Le résultat de la problématique de tri peut proposer que certaines catégories soient vides, par exemple parce que l'ensemble des alternatives à trier ne contient aucune alternative jugée intrinsèquement bonne. Le choix fait intervenir des jugements en général relatifs : il faut être suffisamment bon par rapport aux autres alternatives de l'ensemble considéré pour être retenu comme une alternative de choix potentiel. Mais contrairement au rangement, il est inutile dans cette problématique de ranger toutes les alternatives : sachant qu'il suffit d'en choisir une, on se concentrera sur les candidats potentiels à la tête de liste. Enfin, la problématique de sélection de portefeuilles propose de sélectionner un sous-ensemble d'alternatives, et non une seule. Cette problématique peut être vue comme un problème de tri (ce que nous proposerons), mais à la différence importante que l'ensemble retenu doit respecter certaines propriétés définies sur l'ensemble des alternatives retenu, et non sur les alternatives elles-mêmes.

1.2.4 Remarque concernant les termes alternatives, critères, attributs

Terminons par une remarque concernant le vocabulaire adopté. Le terme *alternative*, en français, désigne un choix entre deux options. Nous l'utilisons pour désigner les objets élémentaires considérés dans le problème de décision.

Cela convient bien dans une problématique de portefeuille. L'alternative désigne alors le choix d'inclure, ou ne pas inclure, l'action qu'elle représente dans le portefeuille. Dans l'exemple proposé ci-dessus, l'action représentée est : visiter tel site. L'alternative est : visiter ou ne pas visiter ce site. Le terme s'applique moins bien lorsque, comme nous le ferons, nous envisageons la problématique de portefeuilles plus généralement comme un problème de partitionnement d'un ensemble d'objets. De plus, ce terme n'est pas toujours adéquat pour les autres problématiques. Dans le cas du tri, les objets à trier ne correspondent pas à des alternatives, car il n'y a pas de problèmes formulés en termes de choix binaires dans une telle situation.

Par uniformité, parce que le mot est souvent employé en ce sens en aide multicritère à la décision, et afin d'éviter l'usage de termes trop généraux tels que « objet », nous utiliserons néanmoins le terme alternative quelle que soit la problématique considérée. De plus, nous visons directement l'enfer sans passer par le purgatoire car même dans le cas de la sélection de portefeuilles, le terme alternative désignera généralement l'objet lui-même (le site, dans l'exemple) et non le choix de son inclusion, ce constitue là encore un abus de langage.

Nous ferons parfois référence au concept d'attribut, à distinguer du critère. Ces deux termes doivent être compris au sens proposé par Roy (1985, p. 228), qui fait remarquer que le mot critère a en français deux sens distincts. Le mot peut faire référence à un caractère qui permet la distinction, sans qu'un jugement de préférence y soit associé, auquel cas nous utiliserons le terme attribut. Ainsi, le sexe d'un candidat peut être un attribut, si le décideur, dans son problème de décision, n'a pas de préférence pour un genre particulier. Le mot critère en français peut également désigner ce qui sert de base à un jugement de préférence, et c'est le sens que nous réservons au terme critère. Un attribut pourra donc être représenté par une fonction $g_j : \mathbb{A} \mapsto X_j$, mais à la différence du critère, ne sera pas associé à une relation de préférence \succeq_j . Les méthodes d'AMCD se préoccupent habituellement uniquement des critères, mais les attributs peuvent occasionnellement être utiles. Dans un problème de portefeuille, le décideur peut s'intéresser à l'équilibre des genres induit par l'ensemble de candidats retenus, auquel cas l'attribut sexe sera utile dans la modélisation du problème.

1.2.5 Notations concernant la problématique de tri

Comme ce travail s'intéresse à la problématique de tri, introduisons quelques notations qui seront utiles dans ce cadre.

L'ensemble de catégories auxquelles les alternatives doivent être affectées est noté \mathcal{C} , une catégorie est $C \in \mathcal{C}$, et nous postulerons un ordre de préférence total sur les catégories. Les catégories peuvent donc être numérotées selon leur

1 Positionnement du travail

position dans l'ordre de préférence (C_1, C_2, C_3 , etc.). Par convention, les catégories sont numérotées dans l'ordre de préférence croissant, C_1 désignant donc la moins bonne catégorie. Il y a n_C catégories. La meilleure catégorie est simplement notée C_n . Pour alléger les notations, nous éviterons l'usage de l'indice associé aux catégories lorsqu'il n'est pas nécessaire. L'ordre de préférence sur les catégories est noté comme suit : $\forall C, C' \in \mathcal{C}, C \triangleleft C'$ indique que C est une catégorie moins bonne que C' , $C \trianglelefteq C'$ indique que C est moins bonne ou égale à C' , $C \triangleright C'$ indique que C est meilleure que C' .

Nous utiliserons parfois des affectations en des intervalles de catégories plutôt qu'en une catégorie. Un intervalle de catégorie $[\underline{C}, \overline{C}]$ est un sous-ensemble de catégories contiguës dans \mathcal{C} :

$$[\underline{C}, \overline{C}] = \{C \in \mathcal{C} \mid \underline{C} \trianglelefteq C \trianglelefteq \overline{C}\}.$$

L'ensemble des intervalles sur \mathcal{C} est noté \mathcal{C}_I , et un intervalle sera parfois noté I lorsque les catégories extrêmes n'importent pas. Plus généralement, l'ensemble des parties de \mathcal{C} est noté $2^{\mathcal{C}}$. On a donc $\mathcal{C}_I \subseteq 2^{\mathcal{C}}$.

Appellons fonction de tri précise une fonction $f : \mathbb{A} \mapsto \mathcal{C}$. Étant donné une telle fonction, la notation $a \in C$ désigne que l'alternative a est affectée à la catégorie C par cette fonction. Nous manipulerons souvent au lieu d'une telle fonction de tri une relation \succeq sur l'espace des alternatives et l'ensemble des catégories : $a \succeq C$ signifie que l'alternative a est affectée au moins à la catégorie C . La relation \succeq , dans le cadre de ce travail, satisfait toujours : $a \succeq C \Rightarrow a \succeq C', \forall a \in \mathbb{A}, C' \trianglelefteq C$. On a nécessairement $a \succeq C_1, \forall a \in \mathbb{A}$. Une telle relation est équivalente à une fonction de tri : $a \in C \Leftrightarrow a \succeq C \wedge [a \not\succeq C', \forall C' \triangleright C]$.

Nous considérons à partir d'ici que nous faisons face à un problème de décision déjà structuré, c'est-à-dire dont les alternatives et critères ont été précisés et la problématique fixée. Le reste du chapitre présente différents objectifs de modélisation et différentes façons d'obtenir des informations concernant les préférences du décideur.

1.3 Approche visant à obtenir un modèle précis

Les approches proposées en aide multicritère à la décision ont souvent pour objectif de déterminer un ensemble de réponses que nous appellerons ici « complet », ou de manière équivalente un modèle appelé précis. Dans une problématique de rangement, cela signifie que les résultats permettent de déterminer de manière univoque l'ordre de préférence de toutes les alternatives considérées. Dans une problématique de choix, les résultats indiqueront soit une seule alternative jugée sans conteste meilleure que toutes les autres, soit un sous-ensemble d'alternatives qu'on affirmera toutes également valables et meilleures

que toutes les alternatives de l'ensemble complémentaire. Dans une problématique de tri, les résultats indiqueront précisément à quelle catégorie chaque alternative appartient. Par comparaison, un ensemble de réponses incomplet laisse, dans une problématique de rangement, certaines paires d'alternatives non rangées : les résultats de la procédure d'aide à la décision n'indiquent alors pas comment ces paires se comparent. Dans une problématique de tri, certaines alternatives peuvent être affectées à plus d'une catégorie : les résultats n'indiquent qu'un sous-ensemble de catégories possibles, plutôt qu'une catégorie précise pour chacune. Formellement, dans le cas du tri, il s'agit de la différence entre déterminer une fonction de tri précise $f : \mathbb{A} \mapsto \mathcal{C}$, et déterminer une fonction de tri imprécise $f : \mathbb{A} \mapsto 2^{\mathcal{C}}$.

Une des méthodes utilisées classiquement pour modéliser les préférences dans le cadre de l'approche visant à obtenir des réponses complètes se base sur la théorie de la valeur (Keeney et Raiffa, 1976). Nous exposerons son fonctionnement technique dans le cadre spécifique de la problématique de tri à la section 2.1.1. Nous nous concentrons pour le moment sur la philosophie sous-jacente.

Cette approche semble s'imposer naturellement : quel meilleur moyen d'éclairer le décideur qu'en lui fournissant des réponses univoques et complètes ? Cependant, de nombreuses études ont mis au jour des difficultés importantes liées à cette approche et ont conduit certains chercheurs à considérer comme irréalistes ces ambitions de complétude.

1.3.1 Incomplétude des préférences

De nombreuses études ont montré que le décideur ne connaît pas ses propres préférences (Kahneman et Tversky, 2000 ; Lichtenstein et Slovic, 2006a). Un consensus s'est formé progressivement, ces dernières décennies, en faveur de la thèse selon laquelle le décideur construit ses préférences au fur et à mesure de sa confrontation à des situations de décision hypothétiques ou réelles, et non qu'il va chercher les indications de préférences qu'il donne dans un ensemble pré-établi de préférences qui se trouveraient quelque part dans sa tête (Lichtenstein et Slovic, 2006a). Le terme de consensus est surtout valable dans les champs de la psychologie et de l'aide à la décision, il l'est moins pour ce qui concerne les économistes. Mais les rangs des sceptiques se réduisent (Lichtenstein et Slovic, 2006b, p. 6–12). Daniel Kahneman, auteur avec Amos Tversky de nombreuses études qui ont largement contribué à la crédibilité de cette thèse, a reçu le prix nobel d'économie en 2002 pour ces travaux.

En particulier, de nombreuses études ont montré que la façon d'interroger des décideurs pour obtenir des indications concernant leurs préférences peut conduire à une différence systématique dans les réponses des décideurs, alors même qu'il s'agit de formulations différentes de la même question sous-jacente.

1 Positionnement du travail

	Connaissance technique	Relations humaines
Candidat a	86	76
Candidat b	78	91

TABLE 1.1 – Évaluation des candidats indiquées au sujet pour le choix du candidat préféré.

Il n'est généralement pas possible d'écarter une des façons de poser les questions comme étant incorrecte : elles apparaissent toutes aussi correctes l'une que l'autre. En pratique, l'analyste suivant l'objectif de la complétude des réponses est contraint de choisir une façon de poser des questions, ce qui conduit à un modèle de préférence qui est donc en partie arbitraire.

Illustrons avec un exemple d'une telle étude (Tversky *et al.*, 1988). L'étude utilise deux procédures de questionnements utilisées couramment pour calibrer un modèle de préférence dans la théorie de la valeur : le questionnement par choix binaire et le questionnement par ajustement (*matching*). L'énoncé suivant (traduit de l'anglais par nos soins) est donné à chaque sujet d'un premier groupe, indépendamment l'un de l'autre. Cet énoncé implémente une procédure de questionnement par choix binaire. « Imaginez que, en tant que dirigeant d'une entreprise, vous deviez choisir entre deux candidats à un poste d'ingénieur de production. Les candidats ont été interviewés par un comité qui les a évalués sur deux attributs (connaissance technique et relations humaines) sur une échelle allant de 100 (excellent) à 40 (très faible). Les deux attributs sont importants pour le poste en question, mais la connaissance technique est plus importante que les relations humaines. Sur la base des évaluations suivantes, lequel de ces deux candidats choisiriez vous ? » Les évaluations reprises dans la table 1.1 sont montrées au sujet. Soixante-cinq pourcents des sujets ($N=63$) choisissent le candidat a , et trente-cinq pourcents choisissent le candidat b . La plupart des sujets choisissent donc le candidat ayant le plus haut résultat sur l'attribut le plus important, la connaissance technique.

Dans un autre groupe, les sujets sont interrogés par ajustement. Ils reçoivent les mêmes données sauf qu'une des quatre évaluations manque. L'énoncé du problème leur demande de « compléter l'évaluation manquante de façon à rendre les deux candidats également aptes à la tâche ». Par exemple, le sujet est confronté à la table 1.2. L'indication de la valeur manquante permet de déduire quel est le candidat préféré, entre le candidat original a ayant les performances (86, 76) et le candidat original b ayant les performances (78, 91). Par exemple, si le sujet répond que pour lui, une performance de 80 établit l'indifférence entre le candidat $x = (86, 76)$ et le candidat $y = (80, 91)$, cela semble indiquer une préférence du sujet pour a contre b (car b est moins bon que y).

1.3 Approche visant à obtenir un modèle précis

	Connaissance technique	Relations humaines
Candidat x	86	76
Candidat y	?	91

TABLE 1.2 – Évaluation des candidats indiquées au sujet pour l’indication d’une valeur établissant l’indifférence entre les candidats.

Quatre groupes d’environ soixante sujets ont dû répondre à cette tâche, chacun avec une valeur manquante différente. Dans chacun de ces quatre groupes, une minorité de sujets (entre vingt-six et quarante-quatre pourcents) exprime, via la procédure d’ajustement, une préférence en faveur de a . Cela montre que la façon de poser la question influence de façon importante la préférence exprimée par le sujet.

Les études de ce type rendent peu raisonnable la prétention à une représentation précise et complète des préférences d’un décideur dans un modèle formel.

Même en supposant que le décideur aurait des préférences précises en tête, la quête d’un modèle précis serait freinée par un problème pratique. Calibrer précisément un modèle de préférence requiert d’obtenir des réponses à de nombreuses questions exigeantes, cognitivement difficiles. Les décideurs sont des gens occupés, et ont peu de temps à consacrer à l’analyste. Il faut donc en général se contenter d’un faible nombre de réponses, qui réduit le nombre de modèles de préférence compatibles avec les affirmations du décideur, mais sans généralement le réduire à un unique modèle.

Notons que certaines procédures (Greco *et al.*, 2011 ; Bous *et al.*, 2010) proposent, étant donné l’ensemble des modèles de préférence compatibles avec les affirmations du décideur, de choisir le modèle de préférence unique le représentant en sélectionnant le modèle respectant le mieux une certaine propriété, en un certain sens défini rigoureusement, par exemple le modèle qui est le plus éloigné des bords du polyèdre défini par les contraintes sur l’espace des modèles de préférence. Ceci permet éventuellement de rendre le choix du modèle déterministe, mais pas de s’affranchir de son caractère arbitraire, car un autre critère de sélection aurait tout aussi bien pu convenir.

Étant donné les arguments évoqués ici, nous pensons que l’approche visant à obtenir des réponses complètes est nécessairement associée à un certain arbitraire dû aux choix que l’analyste est forcée d’effectuer lors de l’élicitation.

1.3.2 Cas où cette approche est justifiée

Est-ce à dire que cette approche est nécessairement invalide et devrait être abandonnée? Nous ne le pensons pas : dans certaines circonstances, elle est justifiée, car il est parfois justifié que l'analyste effectue un choix arbitraire pour obtenir un modèle de préférence précis et unique. L'approche est particulièrement utile lorsque l'objectif du décideur est d'établir un standard, une norme contraignant les choix futurs. Par exemple, lorsqu'un organisme officiel effectue un appel d'offre, il doit parfois publier par avance la façon dont il va évaluer les offres reçues. Cela permet entre autres d'éviter la corruption. Même s'il y a plusieurs façons raisonnables de choisir une offre parmi l'éventail des offres reçues, la sélection a priori d'un seul modèle de préférence rend la procédure de choix déterministe et reproductible, et transparente si le modèle de préférence a été publié avant la décision. De manière plus générale, cet argument s'applique lorsqu'il s'agit d'allouer des ressources limitées à des personnes d'une façon qui est souhaitée transparente : sélection de candidats lors d'un recrutement, sélection d'étudiants lorsque le nombre de places est limité, attribution d'une prime, etc.

Observons au passage que la publication du modèle de préférence qu'un comité utilisera peut faciliter largement le risque de manipulation : dans un contexte d'appel d'offre par exemple, cela peut permettre aux candidats d'améliorer leur offre sur les critères qui ont été privilégiés dans le modèle de préférence. Au contraire, ne pas publier la méthode de choix oblige les candidats à essayer d'être les meilleurs possibles sur tous les critères qu'ils peuvent imaginer pertinents, ce qui augmentera peut-être la qualité globale des offres reçues. Il s'agit là d'un compromis à effectuer par le décideur. Nous ne nous attarderons pas sur cet aspect des choses dans ce travail.

1.4 Comparaison avec le constructivisme

Étant donné les constatations résumées ci-dessus, un courant a émergé en AMCD qui se donne pour objectif non pas de modéliser des supposées préférences pré-existantes d'un décideur, mais de l'aider à construire ses préférences. Il s'agit de l'approche constructiviste de l'aide à la décision, proposée par Roy (1985, 1990, 1992, 1993).

1.4.1 L'approche constructiviste de l'aide à la décision

Selon le courant constructiviste, il n'est pas nécessaire, voire il est irréaliste, de supposer que le décideur a des préférences que le processus d'élicitation chercherait à découvrir (Roy, 1985). Au contraire, le décideur construit ses

préférences au cours du processus. Selon cette vision, le rôle de l'analyste est de l'aider dans cette tâche de construction de ses propres préférences. En effet, la qualité d'une décision peut être vue comme fonction de la qualité du processus ayant conduit à la décision (Slovic, 1995 ; Sunstein, 1991). Une décision pourra être considérée comme meilleure si elle est prise par un décideur qui a pris conscience des enjeux, connaît bien les alternatives en présence et a longuement réfléchi aux conséquences de ses choix.

Payne *et al.* (1999) proposent un ensemble de mesures qui indiquent la qualité d'une telle construction. Ils proposent par exemple de vérifier que les préférences obtenues soient insensibles à des variations qui ne devraient pas les affecter (telles que la façon de présenter les questions), mais sensibles à des variations qui devraient les affecter (telles qu'un changement important sur les performances d'une alternative). Les auteurs font également remarquer que la plupart des gens considèrent que leur préférence devrait être transitive, ou respecter la dominance.

1.4.2 Deux visions compatibles

La vision que nous exposons dans ce chapitre est quelque peu différente de la vision constructiviste, bien que ces deux visions soient compatibles. En effet, dans notre présentation, le décideur a bien des préférences pré-existantes à l'intervention de l'analyste, et une partie du problème consiste à modéliser correctement ces préférences pré-existantes. Comme le font remarquer Lichtenstein et Slovic (2006a, p. xvi), il y a des limites au constructivisme. Certaines personnes n'achèteraient jamais une voiture de sport, certaines personnes adorent le chocolat. Ce sont là potentiellement des préférences stables. Nous appelons préférence stable une préférence qui ne change pas durant un temps suffisamment grand pour qu'il soit intéressant de la prendre en compte dans un modèle de préférence et d'appuyer des recommandations sur le modèle construit ; et une préférence dont l'expression n'est pas susceptible de changement selon la façon dont la question est posée. Nous n'exigeons donc pas qu'il puisse être déterminé qu'une préférence ne changera jamais pour la qualifier de stable. Observons avec Norton *et al.* (1998) que le concept de préférence stable devrait dépendre de l'horizon considéré dans un contexte de modélisation donné : à très long terme, il est possible qu'à peu près aucune préférence ne soit stable. (Ceci constitue cependant une question de recherche sur laquelle il n'est pas nécessaire de prendre position ici.)

Ce travail se place dans un cadre où l'horizon des recommandations est tel qu'il paraît raisonnable de supposer certaines préférences comme stables, dans le sens défini ici. Lorsque nous parlons de contraintes représentant les préférences du décideur, et de sous-ensembles de modèles compatibles avec les

1 *Positionnement du travail*

préférences du décideur parmi une classe de modèles donnée, c'est à ce concept de préférence stable que nous faisons référence. Nous proposerons à la section 9.2 une définition plus précise de ces préférences stables ainsi qu'une manière de tester les modèles de telles préférences.

La vision classique adoptée par les chercheurs constructivistes en AMCD ne prête pas d'attention, à notre connaissance, à ces préférences stables. Roy (1990, résumé, p. ii) indique que « contrairement à leurs homologues en sciences physiques et naturelles, les concepts, modèles et procédures en usage, en Recherche Opérationnelle et Aide à la Décision, ne peuvent que difficilement prétendre décrire une réalité qui serait indépendante de l'observateur et existeraient indépendamment de la personnalité d'autres acteurs humains. »

Cette position semble interdire la prise en compte de préférences stables. Nous ne prétendons pas que tous les chercheurs constructivistes adoptent cette posture, et nous pensons que la vision constructiviste est compatible avec la prise en compte de préférences stables. On peut souhaiter aboutir à un modèle de préférence précis, choisi parmi l'ensemble des modèles compatibles avec les préférences stables du décideur, modèle de préférence qui représenterait des préférences construites par le décideur au cours du processus d'AMCD.

Il faut noter que les outils présentés dans ce travail n'exigent pas pour être utiles que le lecteur adhère à l'idée que nous proposons ici selon laquelle les préférences stables devraient être prises en compte lors de la construction des modèles de préférence. Le lecteur convaincu par l'intérêt de l'approche constructive, et pas par notre concept de préférence stable, peut s'y tenir : nos outils doivent alors être compris comme des points d'appuis permettant de faciliter la construction des préférences du décideur. Il est connu que des mêmes méthodes d'AMCD peuvent souvent être utilisées, avec des variations, selon plusieurs perspectives (Genard et Pirlot, 2002 ; Perny, 1992, p. 24-25).

Quelle que soit la position épistémologique choisie par le lecteur, il admettra probablement que l'incomplétude des préférences telle qu'exposée ci-dessus entraîne l'impossibilité de déterminer un modèle de préférence précis sans ajouter de choix arbitraires de la part de l'analyste. Pour réduire cette part d'arbitraire, deux approches ont été proposées, que nous appelons approche visant à obtenir un modèle imprécis et approche visant à obtenir un ensemble de modèles. Ces deux approches ont été proposées dans un cadre constructiviste, mais nous proposerons en fin de travail une autre façon de les concevoir.

1.5 Approche visant à obtenir un modèle imprécis

Une approche pour prendre en compte l'incomplétude des préférences consiste à privilégier l'usage de modèles non précis, c'est-à-dire de modèles

1.6 Approche visant à obtenir un ensemble de modèles

ne donnant pas nécessairement de réponses complètes. Roy (1985) a le premier développé cette approche et a proposé une famille de modèles, la famille ÉLECTRE, qui s'inscrit dans cette perspective. Dans un problème de tri par exemple, le modèle ÉLECTRE TRI propose d'affecter une alternative à un intervalle de catégories plutôt qu'à une seule catégorie, si l'on considère la procédure d'affectation optimiste et la procédure d'affectation pessimiste décrites en Section 2.2.2. Un tel modèle imprécis peut donc indiquer une hésitation du décideur, une situation où il n'y a pas suffisamment d'arguments pour trancher clairement. De manière conceptuellement similaire, le rangement proposé par le modèle ÉLECTRE III n'est pas nécessairement complet. Dans une situation où la comparaison de deux alternatives données est difficile, le modèle ne se prononce pas sur la supériorité d'une alternative sur l'autre.

Précisons que l'auteur principal de cette famille de modèle, Bernard Roy, met en garde contre l'idée qu'il serait possible de rendre l'analyste totalement neutre. Pour Roy (1985), l'utilisation d'une méthode imprécise (selon notre terminologie) ne suffit pas à retirer toute part active au rôle de l'analyste. Il s'agirait là, d'après Roy, d'un objectif inaccessible.

L'incomplétude des réponses apportées par ces modèles se base sur la notion centrale d'incomparabilité. Lorsque deux alternatives a et b sont comparées, dans le modèle classique de la théorie de la valeur, seules trois situations sont possibles : a est meilleure que b , a est indifférente à b (et doit donc être considérée comme également valable), ou b est meilleure que a . Une des différences principales entre les modèles basés sur la théorie de la valeur et les modèles de la famille ÉLECTRE est l'introduction d'une quatrième possibilité : deux alternatives sont jugées incomparables s'il n'y a pas d'arguments suffisamment déterminants en faveur de l'une des trois autres options. Le fonctionnement technique de certains de ces modèles sera présenté en Section 2.2.2. Cependant, nous ne nous attardons pas sur cette approche, car cette thèse ne la mobilise pas. Nous nous baserons principalement sur une variante d'ÉLECTRE TRI qui, parce qu'elle n'utilise que l'affectation pessimiste, ne présente pas cette caractéristique d'imprécision.

1.6 Approche visant à obtenir un ensemble de modèles

Une autre façon d'éviter le choix arbitraire d'un modèle parmi la multitude de possibles est devenue populaire plus récemment. L'idée consiste à considérer plusieurs modèles au lieu d'un seul. L'approche vise à assurer une certaine robustesse face à l'indétermination du modèle de préférence, nous l'appellerons

1 Positionnement du travail

donc simplement l'approche robuste (Roy *et al.*, 2007; Vincke, 1999).

Considérons une classe de modèles choisie pour représenter les préférences du décideur et son ensemble de paramètres possibles Ω^* . Les informations de préférence que le décideur indique déterminent des contraintes sur l'ensemble Ω^* . Supposons que la classe de modèles retenue fasse intervenir comme paramètres un poids w_j associé à chaque critère $j \in \mathcal{J}$, et que le décideur dise que le poids d'un critère donné devrait être plus important que le poids de tel autre critère. Cela réduit l'espace des paramètres possibles à un sous-ensemble $\Omega \subset \Omega^*$, mais sans réduire l'espace suffisamment pour déterminer un seul modèle. Au fur et à mesure que le décideur donne des informations de préférence, en supposant qu'elles ne soient pas contradictoires, l'espace des paramètres compatibles, et donc des fonctions de préférence compatibles, se réduit de plus en plus. En règle générale toutefois, pour les raisons indiquées ci-dessus, il n'est pas réaliste de tenter de questionner le décideur jusqu'à obtenir un seul modèle compatible. On peut s'en accommoder, et considérer le sous-ensemble de réponses qui sont valides pour tous les modèles compatibles avec les contraintes spécifiées.

Illustrons plus précisément le fonctionnement de cette approche dans le cas de la problématique de tri. Considérons une classe de modèles imprécis M et son espace de paramètres Ω^* , déterminant donc des fonctions de tri imprécises f_ω , étant donné un ensemble de paramètres $\omega \in \Omega^*$. Dès lors, étant donné un espace de paramètres possibles $\Omega \subseteq \Omega^*$, on définira la fonction de tri robuste $f_\Omega : \mathbb{A} \mapsto 2^{\mathcal{C}}$ comme suit, $\forall a \in \mathbb{A}$:

$$f_\Omega(a) = \bigcup_{\omega \in \Omega} f_\omega(a).$$

Nous utilisons le même symbole f pour désigner ces deux fonctions différentes, l'indice indiquant s'il s'agit d'une fonction de tri déterminée par un modèle $\omega \in \Omega^*$ ou s'il s'agit d'une fonction robuste définie comme ci-dessus.

Ainsi, il est éventuellement possible d'affirmer que telle alternative ira sûrement dans la catégorie C_1 ou la catégorie C_2 , et sûrement pas dans une des autres, étant donné les contraintes spécifiées à un stade donné du processus, car tous les modèles compatibles avec ces contraintes affirment que l'alternative est affectée soit à C_1 , soit à C_2 , mais aucun modèle compatible ne se prononce pour une autre catégorie.

L'approche robuste s'applique à tous types de modèles, qu'ils soient précis ou imprécis, selon la nomenclature proposée précédemment. Notons que la préoccupation de robustesse peut concerner bien d'autres aspects de la modélisation que le choix du modèle de préférence (Roy, 1989, 2005b). Par exemple, les performances des alternatives peuvent être connues imparfaitement, auquel cas l'analyste devra prendre garde à ce que les conclusions apportées soient ro-

bustes face à ce type d'incertitude. Nous n'abordons dans ce document que les aspects de la robustesse concernant les modèles de préférence.

L'approche robuste a été rapidement liée à une proposition permettant de contraindre plus facilement l'espace des modèles, appelée la désagrégation.

1.7 Désagrégation des préférences

Une procédure de désagrégation (Dias *et al.*, 2002 ; Siskos *et al.*, 2005) est une procédure permettant d'obtenir des paramètres d'un modèle d'agrégation en partant d'affirmations de préférences concernant un ensemble d'alternatives utilisées comme exemples. Le modèle choisi est un des modèles qui reproduit les affirmations de préférences données en exemple. La désagrégation procède à l'inverse de l'agrégation, qui supposant connus les paramètres d'un modèle d'agrégation, l'utilise pour déterminer quelles préférences sont valables sur un ensemble d'alternatives. On parle en général de processus d'agrégation / désagrégation, car les deux techniques sont généralement combinées. La désagrégation permet d'obtenir les paramètres du modèle à l'aide d'un ensemble d'alternatives utilisées comme exemples. Cela permet ensuite de procéder à l'agrégation, qui donne les résultats attendus sur un autre ensemble, ou sur un sur-ensemble, d'alternatives.

Illustrons dans le cas de la problématique de tri, en supposant un modèle simple ne faisant intervenir comme paramètres que des poids, associés à chaque critère, et des seuils numériques séparant les catégories. Selon un procédé d'élicitation classique, le décideur pourrait être interrogé directement sur les paramètres qu'il estime adéquats. Une fois les poids et les seuils déterminés, le modèle indiquerait, pour toute alternative, la catégorie à laquelle elle serait affectée. Alternativement, avec un modèle imprécis, le modèle déterminerait dans quel sous-ensemble de catégories l'alternative peut aller. La désagrégation, elle, propose une manière d'aider le décideur à fixer les paramètres, en lui proposant de fournir des exemples d'affectation. Un exemple d'affectation est l'indication d'une alternative, avec ses performances sur les critères du problème, et la catégorie à laquelle cette alternative devrait être affectée selon le décideur. Notons $X \subset \mathbb{A}$ l'ensemble des alternatives utilisées en exemples, et $C_x \in \mathcal{C}$ la catégorie à laquelle le décideur indique que l'alternative $x \in X$ devrait être affectée. Définissons maintenant l'ensemble des modèles d'une classe donnée satisfaisant ces exemples d'affectation : il s'agit des modèles qui affectent chaque alternative utilisée en exemple à la catégorie choisie par le décideur pour cet exemple. Formellement, cet ensemble de modèles Ω est défini comme suit (en supposant une classe de modèles précis associée à un ensemble de paramètres Ω^*) :

$$\Omega = \{\omega \in \Omega^* \mid f_\omega(x) = C_x, \forall (x, C_x) \in X\}.$$

1 Positionnement du travail

Cet ensemble de modèles détermine un ensemble de candidats à la représentation des préférences du décideur. Si l'approche visant à obtenir un modèle précis est adoptée, il suffit ensuite de choisir un des modèles de l'ensemble obtenu, ou de continuer à contraindre l'ensemble en ajoutant des exemples.

Les procédures de désagrégations les plus connues, UTADIS (Devaud *et al.*, 1980) dans le cadre du tri et UTA (Jacquet-Lagrèze et Siskos, 1982) dans le cadre d'une problématique de rangement (Siskos *et al.*, 2005), proposent de choisir un modèle parmi les modèles compatibles. Par la suite, cette idée fut combinée avec celle de robustesse exposée ci-dessus (Dias *et al.*, 2002). Dans ce cas, l'ensemble complet des modèles compatibles avec les exemples donnés par le décideur est utilisé pour obtenir des réponses.

Dans ce travail, nous proposerons plusieurs procédures de désagrégation basées sur la classe de modèles ÉLECTRE TRI. Certaines proposent ensuite de tirer des conclusions robustes, donc basées sur l'ensemble entier de modèles compatibles avec les contraintes imposées par les affirmations du décideur, d'autres propositions viseront à obtenir un modèle unique de tri.

Il nous reste à présenter une facette importante de toute procédure d'aide multicritère à la décision, à savoir le traitement des inconsistances.

1.8 Traitement des inconsistances

Jusqu'à présent, nous avons supposé dans nos descriptions que le décideur est parfaitement en phase avec la classe de modèles choisis. Mais il peut arriver, en particulier dans un cadre de désagrégation (nous ne parlerons que de ce cas) que les affirmations du décideur ne soient pas représentables avec un modèle de la classe choisie. Dans une problématique de tri, un décideur pourrait donner des exemples d'affectation tels qu'aucun modèle de la classe choisie ne puisse les représenter. Une signification possible est que la classe de modèles choisie n'est pas adéquate pour représenter les raisonnements du décideur. Mais ce n'est pas la seule conclusion possible. Il est également intéressant d'expliquer cette incompatibilité entre l'ensemble d'exemples et la classe de modèles (ou une certaine propriété de cette classe de modèles), afin de demander au décideur ce qu'il souhaite changer. Peut-être souhaite-t-il adopter un raisonnement compatible avec la classe de modèles choisie, même s'il lui faut pour cela changer certains exemples. Une violation de la dominance est un exemple extrême d'une telle situation : si l'analyste fait remarquer au décideur que certains de ses exemples ne respectent pas la propriété de dominance, le décideur préférera sans-doute modifier ses exemples que déclarer que la violation de la dominance ne le dérange pas. Dans d'autres cas, l'inverse pourrait se produire : le décideur pourrait affirmer que ses exemples sont corrects, auquel cas la classe de modèles

choisie pour représenter les préférences du décideur devrait être remplacée par une autre, à même de respecter ces exemples.

Pour certaines classes de modèles, des outils permettent de détecter les inconsistances en exhibant un sous-ensemble d'exemples qui posent problème. Notons qu'il existe en général plusieurs façons de résoudre l'inconsistance, plusieurs sous-ensembles d'exemples dont la modification permet de restaurer l'existence d'un modèle compatible. Une stratégie possible est de chercher un sous-ensemble d'exemples à modifier aussi petit que possible. Nous verrons au chapitre suivant des méthodes visant à traiter les problèmes d'inconsistance.

1.9 L'AMCD pour la décision de groupe

L'aide à la décision peut viser à aider un groupe de décideurs, plutôt qu'un seul décideur. Une manière évidente d'aborder une telle situation est de considérer le groupe de décideurs comme une seule personne, auquel cas les approches citées précédemment s'appliquent. Cette stratégie simple fonctionne particulièrement bien lorsque les décideurs ont des raisonnements très proches. Mais les décideurs peuvent parfois avoir des idées différentes quand à la façon d'évaluer les alternatives, par exemple parce qu'ils n'accordent pas la même importance à tous les critères. Distinguer les décideurs dans la procédure d'aide à la décision peut alors apporter une plus grande aide.

1.9.1 Positionnement du travail

L'aide à la décision appliquée aux groupes de décideurs, dans sa dimension générale, est extrêmement large. Matsatsinis et Samaras (2001) proposent une importante revue de littérature, avec emphase sur la décision de groupe s'appuyant sur des modèles d'AMCD. Beaucoup de travaux considèrent un cadre non interactif dans lequel les décideurs présentent le rangement des alternatives qu'ils jugent adéquat, ou d'autres informations concernant leurs préférences, et où la méthode consiste à proposer une décision de compromis en agrégeant les différentes informations préférentielles. Certains de ces travaux intègrent explicitement des modèles multicritères (Hochbaum et Levin, 2006). Nous nous concentrons dans ce travail sur les méthodes interactives, prenant en compte des critères conflictuels, et basées sur des représentations imprécises des préférences.

Nous supposons donc, comme dans la situation du décideur unique, que les décideurs sont disponibles pour réagir aux suggestions de l'analyste. L'aide à la décision a alors pour objectif de faire aboutir le groupe à un compromis ou à un consensus concernant un modèle de préférence qui pourra par la suite être

1 Positionnement du travail

considéré comme représentant le groupe. Les méthodes qui nous intéressent s'appuient sur l'approche robuste décrite précédemment : la méthode ne demande pas à chaque décideur de cerner complètement un modèle de préférence précis, mais associe à chaque décideur un ensemble de contraintes sur l'espace des modèles de préférence candidats à le représenter. L'analyse de ces ensembles permet par exemple (selon les méthodes) de diriger la discussion entre les décideurs, ou de tirer des conclusions partielles.

Comme le remarquent plusieurs chercheurs (Dias et Clímaco, 2005), une telle approche tirant parti de données de préférence imprécises s'applique bien à la situation d'une décision de groupe : elle est moins exigeante pour les décideurs, et cette imprécision peut permettre d'exhiber des propositions de décisions contredisant peu les affirmations des décideurs.

Ces approches combinent souvent la robustesse avec des techniques de désagrégation, ce qui permet aux décideurs d'exprimer les contraintes sur leurs modèles individuels de préférence de manière indirecte par le biais d'affirmations de préférences holistiques.

Précisons que nous nous intéressons à une situation où les décideurs visent à aboutir à une décision en collaboration. Ils ne sont pas nécessairement d'accord entre eux au départ, mais sont prêts à échanger leurs points de vue et éventuellement à changer d'avis. Ce n'est donc pas un contexte de négociation, où chaque décideur pourrait avoir pour objectif de tirer la couverture à lui.

1.9.2 Travaux proches

Citons quelques exemples significatifs de travaux s'inscrivant dans la perspective décrite ici.

Greco *et al.* (2009, 2012) proposent des procédures qui, dans le cadre du choix, du rangement, ou du tri, analysent quelles conclusions sont nécessairement vraies, à partir d'informations de préférence holistiques fournies par les décideurs. Un exemple d'une telle conclusion est que telle alternative est nécessairement préférée à telle autre, c'est-à-dire que les informations de préférence données par tous les décideurs aboutissent à cette conclusion. Les outils proposés permettent en outre d'obtenir des informations qui sont sûres pour certains décideurs seulement, ou qui sont possibles pour certains ou tous les décideurs.

Salo (1995) s'intéresse à la problématique de choix et se base sur des modèles individuels basés sur la théorie de la valeur. Les décideurs individuels contraignent leur modèle par des affirmations sur les paramètres des modèles de préférence. Des conclusions robustes sont déduites de ces espaces contraints de manière à déterminer l'alternative à choisir, ce qui peut conduire à un choix consensuel. Sinon, un modèle global représentant tous les décideurs à

l'aide d'une somme pondérée des valeurs fournies par chaque décideur peut être utilisé pour obtenir des conclusions partielles. Des indicateurs sont fournis qui permettent par exemple de déterminer l'intervalle de valeurs qu'une alternative peut mériter d'après le groupe, de manière à diriger et faciliter la discussion.

Dias et Clímaco (2005) suivent une démarche similaire, sans utiliser de désagrégation, et proposent également des outils permettant aux décideurs individuels de comparer leur point de vue au point de vue du groupe. Leur approche est comparée à une série d'études se plaçant dans le même cadre. Ils distinguent également l'agrégation au niveau des espaces des modèles de préférence individuels et l'agrégation au niveau des résultats des modèles individuels et présentent une discussion intéressante de cette différence d'approche.

1.9.3 Traitement des incohérences dans le cas de la décision de groupe

De manière similaire au cas mono-décideur, un problème de cohérence peut se poser lors du processus consistant à contraindre de plus en plus fortement l'ensemble des modèles de préférence compatibles, en vue d'obtenir des conclusions suffisamment précises. Le problème se pose cependant de plusieurs façons. Il se peut que les différents éléments indiqués par un décideur donné ne puissent être représentés dans la classe de modèles choisie. Ce cas ressemble au problème qui se pose dans une situation impliquant un seul décideur et peut être résolu de manière similaire. Il se peut également que chaque décideur soit cohérent en ce qui concerne ses affirmations propres, mais qu'il n'existe pas de modèle commun à tous les décideurs, donc que les affirmations, une fois mises en commun, ne soient pas représentables. Là encore, les outils proposés dans le cas mono-décideur peuvent s'appliquer : on peut chercher un ensemble minimal d'affirmations à retirer ou à modifier pour restaurer la cohérence. Mais on peut aussi désirer assurer un minimum d'équité, c'est-à-dire éviter les solutions qui consistent à demander à certains décideurs de modifier fortement leurs affirmations de préférence alors que d'autres n'ont rien à modifier.

Il est aussi possible de tenter d'éviter l'apparition d'incohérences, au lieu de chercher à les résoudre après coup. Salo (1995) propose de calculer, étant donné une nouvelle affirmation de préférence à introduire dans l'ensemble de contraintes, l'intervalle de valeurs acceptables pour le paramètre concerné par la nouvelle affirmation étant données les affirmations précédentes. Tant que les décideurs restent dans les intervalles recommandés, aucune incohérence ne peut être introduite dans les modèles de préférence ainsi construits. Notons qu'il est vraisemblable que dans des situations pratiques dont la complexité

1 *Positionnement du travail*

justifie une aide formelle, un décideur finisse par se prononcer en faveur d'une valeur en dehors de l'intervalle des valeurs qui conservent la cohérence. Le cas contraire indiquerait une situation de décision peu conflictuelle. Cet argument intuitif devrait cependant être confirmé par des expériences. Par ailleurs, cette stratégie entraîne un risque que l'historique du processus d'aide à la décision influence l'évolution du modèle, ce qui peut être considéré comme un inconvénient dans certaines situations.

Nous examinerons avec plus de détails quelques méthodes plus proches de nos propositions à la section 2.3.

1.10 Motivation et contenu du travail

1.10.1 Motivation

La science a montré au fil de l'histoire sa capacité à aider les hommes face aux problèmes bien définis. De nombreux travaux permettent de savoir comment calculer le plus court chemin entre deux points, réduire le coût de stockage dans une entreprise, produire des ordinateurs plus rapides, fabriquer des machines adéquates pour explorer les océans... Lorsque de tels problèmes sont étudiés, ils sont décrits comme des problèmes bien définis, du point de vue scientifique. Si différentes solutions sont concevables pour résoudre le problème, il est possible de les comparer afin de déterminer la ou les meilleures.

Bien souvent cependant, plus d'une façon de mesurer la qualité de ces solutions potentielles conviendrait. C'est le cas dans le problème de la machine idéale pour l'exploration sous-marine. Une mesure peut alors être choisie par le chercheur, agrégeant les différents critères d'évaluation pertinents d'une certaine manière. Cela permet de se ramener à un problème d'optimisation ayant une solution objective. Mais le choix de la mesure de qualité lui-même est subjectif, ou au moins arbitraire. Une pratique habituelle pour éviter toute critique est de définir le problème de manière plus spécifique dès le départ, par exemple, en prétendant que le problème consiste à construire la machine la plus résistante possible, ou la plus maniable possible.

Cette pratique conduit à apporter des éclairages importants pour guider les choix humains, élargir nos connaissances et améliorer notre vie en société. Mais elle met un aspect important du problème sous le tapis. On ne veut pas simplement la machine la plus résistante possible, tout comme on ne veut pas seulement des ordinateurs les plus rapides possibles : on veut des ordinateurs « meilleurs ». Ce qui doit prendre en compte leur coût, leur fiabilité, voire les préoccupations écologiques ou éthiques.

Une telle formulation vague du problème le rend plus réaliste. Mais cette for-

mulation requiert de préciser comment l'agrégation se fait afin de déterminer quand un ordinateur est meilleur qu'un autre. Cela introduit un aspect subjectif au problème, puisque différents individus auront probablement parfois des avis différents sur la question.

Nous avons dans ce chapitre distingué les préférences spontanées, observables via le comportement d'achat habituel des individus, des préférences réfléchies. Ces dernières sont celles que les individus expriment lorsqu'ils prennent en compte différentes façons de voir le problème, et réfléchissent aux propriétés qu'ils veulent voir respecter par leur méthode de choix. Nous définirons plus précisément cette notion à la section 9.2, mais l'intuition devrait suffire pour le moment pour distinguer ces deux modes de réflexion.

La science a longtemps négligé les aspects subjectifs dans ses modèles, de peur peut-être de perdre le caractère objectif de ses affirmations. C'est encore plus frappant lorsqu'on parle des préférences réfléchies, plus difficilement observables. Les préférences des individus, y compris leurs préférences réfléchies, font pourtant partie de la réalité qui compose notre univers, et nous pensons qu'il est possible d'en dire des choses objectives en observant leurs propos et leur comportement.

L'AMCD propose une voie qui pourrait conduire à une meilleure compréhension des problèmes que nous avons à résoudre, individuellement et collectivement.

1.10.2 Contenu du travail

Ce travail vise principalement à proposer des méthodes d'élicitation, donc des façons de questionner un décideur ou un groupe de décideurs pour obtenir un ou plusieurs modèles de préférence. Ces méthodes utilisent des techniques de désagrégation.

Le chapitre 2 décrit différentes classes de modèles, avec pour certaines leur axiomatique, ce qui permet de les comparer. Il décrit en particulier la classe de modèles que nous utiliserons principalement, MR Sort, et la compare à d'autres modèles proches. Le chapitre reprend aussi en détail certaines méthodes proposées dans la littérature, sur lesquelles nos outils s'appuient ou qui sont proches de nos propositions.

Aspects méthodologiques

Les aspects méthodologiques du travail sont regroupés en partie II.

Le chapitre 3 présente une méthode permettant de faire converger un groupe de décideurs vers un modèle de tri unique. Elle utilise la désagrégation : les décideurs fournissent des exemples non nécessairement consensuels. La méthode

1 *Positionnement du travail*

fournit une manière pour les décideurs de discuter de ces exemples tout en convergeant vers un modèle de préférence. Elle s'appuie sur la formulation de plusieurs Programmes Mathématiques (PMs). Nous présentons également un algorithme pour comparer deux modèles de tri.

Le chapitre 4 introduit une manière d'interroger le décideur d'une façon qui permet de prendre en compte ses hésitations lorsqu'il fournit des exemples d'affectation en vue de paramétrer un modèle de tri. La méthode lui permet d'associer les catégories qu'il propose pour les alternatives à des degrés de crédibilité. Les résultats de la méthode permettent au décideur de visualiser les compromis possibles entre la crédibilité et la précision des conclusions obtenues, sans exiger de sa part un effort beaucoup plus important concernant les informations de préférences à fournir. La méthode proposée permet de réduire l'inconvénient de l'approche robuste consistant en une possible imprécision trop grande des réponses, tout en en gardant les avantages liés à la réduction de l'arbitraire.

Le chapitre 5 propose une méthode d'évaluation de portefeuilles. Elle intègre des préoccupations d'évaluation absolue, afin de s'assurer de la qualité intrinsèque des alternatives sélectionnées, et des préoccupations portant sur l'équilibre du portefeuille résultant. Pour éliciter les préférences du décideur concernant l'évaluation intrinsèque, la méthode s'appuie sur la désagrégation. Les exemples fournis ne définissent pas un modèle de préférence précis, et cette imprécision est utilisée pour déterminer un portefeuille qui satisfait également des contraintes d'équilibre.

Aspects concernant l'implémentation

La partie III propose des contributions de nature plus techniques. Les contributions présentées dans cette partie ont pour objectif de faciliter la mise en œuvre des méthodes proposées, et plus généralement de faciliter la pratique de la modélisation formelle des préférences.

Le chapitre 6 introduit plusieurs concepts proposés par le consortium Decision Deck (<http://www.decision-deck.org/>), un consortium visant à développer des outils open source pour la mise en œuvre de méthodes d'AMCD. Le chapitre décrit les composants logiciels réutilisables que nous avons soumis à la plateforme de services web, composants implémentant plusieurs méthodes d'agrégation et de désagrégation.

Le chapitre 7 propose un modèle de données en vue de faciliter l'échange de données de méthodes d'AMCD entre composants logiciels. Un schéma de données est actuellement utilisé pour normer l'encodage de telles données, mais notre proposition vise à résoudre un certain nombre d'inconvénients dans la façon dont le modèle actuel est implémenté. Bien que ce genre de travaux

soient malheureusement parfois considérés comme trop appliqués pour mériter une grande attention, ils nous semblent au contraire mériter largement qu'on s'y intéresse pour plusieurs raisons. La tâche est difficile, preuve en est que notre proposition vise à améliorer une première version qui représentait déjà un gros travail, alors qu'elle est encore largement perfectible. L'exercice est intéressant, car il oblige à se doter d'un cadre assez général pour pouvoir représenter une gamme de méthodes importante. Enfin, la valeur pratique nous semble importante. Obtenir un bon modèle de données pourrait avoir des répercussions intéressantes sur la compréhension du champ de l'AMCD du point de vue du scientifique, et des répercussions importantes sur la pratique des méthodes.

Le chapitre 8 donne des détails concernant l'implémentation de plusieurs PMs présentés au chapitre 3, détails importants pour garantir que les méthodes soient implémentables et les tests reproductibles.

Conclusions et perspectives

Enfin, le chapitre 9 propose des perspectives qui nous tiennent à cœur pour le champ de l'AMCD.

2 Revue de modèles de tri et de méthodes de désagrégation

Dans ce chapitre, nous présentons plusieurs classes de modèles de tri connues. La dernière famille de modèles présentée est la plus proche de celle que nous utilisons. La dernière section présente en détails quelques méthodes de désagrégation proches de celles que nous développerons ou utiles à notre propos.

Suivant une approche axiomatique classique, nous nous intéressons aux conditions nécessaires et suffisantes pour qu'un modèle puisse représenter une fonction de tri précise d'un ensemble d'alternatives \mathbb{A} vers un ensemble de catégories \mathcal{C} . Ce cadre permet de comparer les modèles : nous verrons que plusieurs d'entre eux entretiennent des relations surprenantes. Certains modèles que nous présentons n'ont toutefois pas été axiomatisés.

Classiquement, les approches de l'AMCD sont présentées par familles de méthodes, mais sans se restreindre à la problématique de tri. Deux familles sont généralement mises en avant (Belton et Stewart, 2002 ; Vincke, 1998). La famille de la théorie de la valeur attribue une valeur à chaque alternative, et les préférences sont déterminées en fonction de ces valeurs. La famille du surclassement procède en comparant les alternatives par paires. Ces deux familles proposent des variantes pour les trois problématiques classiques : ranger, choisir, ou trier.

Nous avons pris le parti de nous focaliser sur les modèles de tri, ce qui nous permet de nous concentrer sur la comparaison entre modèles sur une base axiomatique. En particulier, nous verrons que le modèle de tri basé sur l'intégrale de Sugeno est équivalent au modèle Non Compensatory Sorting Model (NCSM) sans véto, ce qui peut être jugé surprenant car ces modèles semblent utiliser des principes différents. L'axiomatisation du modèle de tri basé sur l'intégrale de Sugeno est connue depuis longtemps (Słowiński *et al.*, 2002), et le résultat d'équivalence avec le modèle NCSM est connu depuis plusieurs années (Bouyssou et Marchant, 2007a,b). Pourtant, ces résultats sont à notre avis trop peu utilisés dans la communauté de l'AMCD. Le résultat d'équivalence mentionné ici est particulièrement intéressant car le modèle de tri basé sur l'intégrale de Sugeno semble intuitivement être plus proche de la famille de la théorie de la valeur, alors que le modèle NCSM, même dans sa version sans véto, semble plus

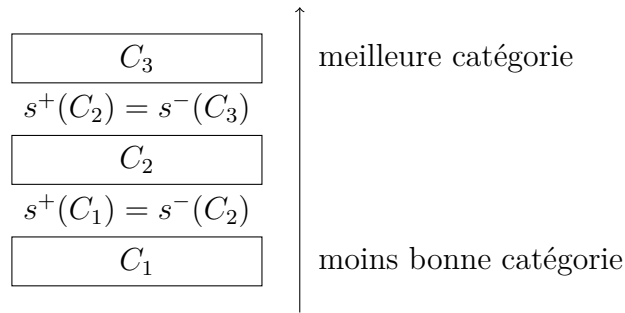


FIGURE 2.1 – Trois catégories et les seuils qui les séparent.

proche des principes du surclassement.

La section 2.1.1 présente le modèle de tri à fonction croissante et à seuils. Il s'agit du modèle le plus général auquel nous nous intéresserons. Un axiome de base est présenté, suffisant à garantir la représentation des préférences dans ce modèle lorsque certaines conditions structurelles sont satisfaites. La section 2.1.2 présente un modèle tout aussi général, mais présenté sous forme de règles. Le modèle de tri à base d'intégrale de Sugeno est présenté à la section 2.1.3. La section 2.2 présente le modèle ÉLECTRE TRI et plusieurs variantes de celui-ci. Parmi eux, le modèle MR Sort nous intéresse particulièrement, car c'est celui sur lequel l'essentiel de ce travail se base. Le modèle NCSM est également particulièrement intéressant car c'est la variante d'ÉLECTRE TRI qui a été axiomatisée, et qui permet donc la comparaison aux modèles présentés précédemment.

2.1 Classes de modèles de tri non basées sur le surclassement

2.1.1 Modèle de tri à fonction croissante et à seuils

Une idée naturelle pour trier des alternatives dans des catégories ordonnées en fonction de leurs performances consiste à utiliser une fonction qui associe à chaque alternative une valeur numérique, et qui affecte une alternative à une catégorie en fonction de seuils numériques.

Associons à chaque catégorie $C \in \mathcal{C}$ un seuil inférieur $s^-(C) \in \mathbb{R}$. Notons $s^+(C)$ le seuil supérieur correspondant à C , égal au seuil inférieur de la catégorie qui lui est immédiatement préférée. La figure 2.1 montre la relation entre les seuils et les catégories.

2.1 Classes de modèles de tri non basées sur le surclassement

Cherchons à attribuer une valeur numérique à chaque alternative à l'aide d'une fonction $v : \mathbb{A} \mapsto \mathbb{R}$, et de seuils numériques $s^-(C) \in \mathbb{R}$, de telle sorte que les valeurs des alternatives et les seuils représentent la préférence du décideur. L'objectif est de trouver une fonction v et des valeurs de seuils tels que $v(a) \geq s^-(C) \Leftrightarrow a \succeq C$, ou de manière équivalente, $s^-(C) \leq v(a) < s^+(C) \Leftrightarrow a \in C$. De plus, on souhaite que la valeur attribuée par v ne dépende que des performances de l'alternative. Pour qu'il existe une telle fonction v et des valeurs s représentant les préférences des décideurs, il faut que les préférences des décideurs ne dépendent elles aussi que des performances des alternatives :

$$[a_j = b_j, \forall j \in \mathcal{J}] \Rightarrow a \sim b, \quad (\text{critères exhaustifs})$$

où $a \sim b$ indique une affectation de a à la même catégorie que b . Si cette condition est satisfaite, nous dirons que les critères sont exhaustifs, car elle est respectée ssi l'ensemble des critères choisis permet d'expliquer la préférence du décideur.

Dans le modèle présenté ici, la valeur d'une alternative est attribuée en deux phases. Premièrement, l'alternative se voit attribuer une valeur partielle pour chacune des dimensions intervenant dans le jugement. Ces valeurs partielles sont notées $v_j(a_j), \forall j \in \mathcal{J}$. Ces valeurs sont ensuite agrégées. Notre fonction s'écrit donc ainsi :

$$v(a) = V(v_1(a_1), v_2(a_2), \dots, v_n(a_n)),$$

que nous écrirons de façon plus courte :

$$v(a) = V(v_j(a_j), j \in \mathcal{J}). \quad (\text{fonction de valeur})$$

Les fonctions $v_j : X_j \mapsto \mathbb{R}, j \in \mathcal{J}$ sont des fonctions de valeurs partielles. La fonction V est appelée fonction d'agrégation. Dans le cas où les ensembles X_j ont au plus la cardinalité de \mathbb{R} , il est clair qu'il est possible de représenter les préférences du décideur dans ce modèle ssi les critères sont exhaustifs. Autrement dit, il est possible de définir, ssi les critères sont exhaustifs, des fonctions v_j , une fonction V , des seuils $s^-(C)$ tels que $v(a) = V(v_j(a_j), j \in \mathcal{J}) \geq s^-(C) \Leftrightarrow a \succeq C$.

Mais pour que ce modèle soit intéressant, il faut s'assurer que les fonctions de valeurs partielles traduisent effectivement la notion de valeur, c'est-à-dire qu'accroître la valeur d'une fonction v_j , toutes choses égales par ailleurs, accroisse la préférence du décideur. Imposons donc que la fonction V soit strictement croissante en chacun de ses arguments et voyons quelles conditions il faut ajouter concernant les préférences du décideur de manière à garantir la représentabilité dans ce modèle à fonction croissante et à seuil. Cela requiert d'utiliser des notations concernant les performances sur des sous-ensembles de critères, nous renvoyons le lecteur à la page xvi pour leurs définitions.

Caractérisation du modèle de tri à fonction croissante et à seuils

Définissons la propriété linear- j : $\forall x_j, y_j \in X_j, C_h, C_i \in \mathcal{C}, a_{-j}, b_{-j} \in X_{-j}$,

$$\left. \begin{array}{l} (x_j, a_{-j}) \prec C_i \\ \text{et} \\ (y_j, a_{-j}) \in C_i \\ \text{et} \\ (x_j, b_{-j}) \in C_h \end{array} \right\} \Rightarrow (y_j, b_{-j}) \succeq C_h. \quad (\text{linear-}j)$$

Intuitivement, cette propriété permet de définir des niveaux de préférence sur le critère j indépendamment des évaluations sur les autres critères. La propriété indique en effet que si le remplacement de x_j par y_j améliore l'alternative lorsque les autres évaluations (sur X_{-j}) valent a_{-j} , alors remplacer également x_j par y_j avec d'autres évaluations valant b_{-j} ne peut pas diminuer l'intérêt pour l'alternative.

La relation de préférence \succeq sur $\mathbb{A} \times \mathcal{C}$ est dite linéaire ssi elle satisfait (linear- j) pour tous $j \in \mathcal{J}$.

Si la relation de préférence \succeq est linéaire, et que des conditions d'ordre-densité sur les ensembles X_j sont satisfaites, alors il est possible de représenter \succeq dans le modèle à fonction croissante et à seuils. Ces conditions sont suffisantes pour garantir la représentabilité dans le modèle, et sont également nécessaires. Les conditions d'ordre-densité exigent que les ensembles X_j ne soient pas « trop grands », nous ne les détaillons pas (l'ensemble des réels, par exemple, ou tout ensemble fini, satisfont la condition). Dans la suite de ce chapitre, nous omettrons ces conditions.

La propriété (linear- j) peut aussi être vue d'une autre manière. Définissons des relations \succeq_j désignant le fait de pouvoir remplacer une évaluation y_j de n'importe quelle alternative $a = (y_j, a_{-j})$ par une évaluation x_j en étant assuré que le changement ne décroît pas la qualité de l'alternative, donc que $a' = (x_j, a_{-j})$ n'est pas affectée à une catégorie pire que a . Les relations \succeq_j sur chaque X_j sont définies comme suit. $\forall x_j, y_j \in X_j$:

$$x_j \succeq_j y_j \Leftrightarrow \forall a_{-j} \in X_{-j}, \forall C \in \mathcal{C}, (y_j, a_{-j}) \in C \Rightarrow (x_j, a_{-j}) \succeq C.$$

Notons que la relation \succeq_j est transitive par construction. La condition (linear- j) est satisfaite ssi \succeq_j est complète. L'existence du modèle est donc garantie ssi les relations \succeq_j sont toutes complètes. De plus, dans ce cas, les fonctions de valeur partielle v_j peuvent toujours être définies de façon à ce qu'elles représentent numériquement les relations \succeq_j , c-à-d $x_j \succeq_j y_j \Leftrightarrow v_j(x_j) \geq v_j(y_j)$. Plus précisément, ces fonctions satisfont nécessairement $x_j \succ_j y_j \Rightarrow v_j(x_j) > v_j(y_j)$.

2.1 Classes de modèles de tri non basées sur le surclassement

Dans ce modèle, nous avons exigé une fonction V strictement croissante en chaque argument. Le modèle supposant une fonction V non décroissante en chacun de ses arguments est équivalent : le deuxième existe ssi le premier existe.

Ce modèle est le modèle le plus général auquel nous nous intéressons dans le cadre de cette thèse.

Notons qu'une variante de ce modèle est souvent utilisée dans le cadre du rangement : au lieu de comparer la valeur des alternatives à des seuils, la valeur peut être utilisée pour déterminer un ordre total sur les alternatives.

Modèle UTADIS

Une variante bien connue du modèle à fonction croissante et à seuils est utilisée dans le cadre de la méthode UTADIS (UTilités Additives DIScrimnantes (Devaud *et al.*, 1980)). Cette méthode propose d'inférer une fonction de tri en utilisant la désagrégation. Nous appelons modèle UTADIS, par abus de langage, le modèle utilisé par la méthode UTADIS.

Le modèle UTADIS utilise une fonction de valeur (voir (fonction de valeur) page 37), où la fonction d'agrégation V est une somme :

$$v(a) = V(v_j(a_j), j \in \mathcal{J}) = \sum_{j \in \mathcal{J}} v_j(a_j).$$

Fixer le modèle requiert de déterminer les fonctions de valeur partielles croissantes v_j et un ensemble de seuils $s^-(C), C \in \mathcal{C} \setminus C_1$. Les alternatives sont ensuite affectées, comme pour le modèle à fonction croissante et à seuils, en fonction de la comparaison de la valeur de l'alternative aux seuils.

Plusieurs travaux ont décrit l'axiomatique de cette classe de modèles (Bouysou et Marchant, 2008, 2009, 2010, 2011). Elle est cependant plus complexe que les axiomatisations présentées dans le reste de cette partie car elle requiert différentes analyses selon différents cas.

La section 2.3 présente la méthode de désagrégation UTADIS.

2.1.2 Tri à base de règles

Le tri peut aussi opérer suivant une logique consistant à suivre un ensemble de règles, ce que propose le modèle basé sur les ensembles approximatifs proposé par Greco *et al.* (2002), ou Dominance-based Rough Set Approach (DRSA). Nous le désignerons par « modèle DRSA ».

Les règles reposent sur un ordre de préférence \succeq_j associé à chaque critère $j \in \mathcal{J}$. Une règle $r = (r_a, r_c)$ est une paire composée d'un antécédent et d'un conséquent. L'antécédent associe chaque critère j d'un sous-

ensemble de critères $J \subseteq \mathcal{J}$ à un seuil $s_j \in X_j$. Par exemple, l'antécédent $r_a = \{(j_1, 3); (j_3, \text{« bleu »})\}$ associe le critère j_1 au seuil 3 et le critère j_3 au seuil « bleu ». Ceci suppose que X_{j_1} est numérique et X_{j_3} représente des couleurs sur lesquelles un ordre de préférence a été défini. Une alternative satisfait l'antécédent de la règle r ssi ses performances dépassent les seuils minimaux requis par l'antécédent : $\forall (j, s) \in r_a, a_j \succeq_j s$. Le conséquent de la règle est une catégorie $r_c \in \mathcal{C}$. La règle indique que les alternatives qui sont suffisamment bonnes pour satisfaire l'antécédent r_a sont affectées au moins à la catégorie r_c .

Un modèle DRSA est paramétré par un ensemble de règles, et trie les alternatives de la façon suivante. Une alternative est affectée à la meilleure catégorie indiquée comme conséquent d'une règle, parmi les règles dont l'alternative satisfait l'antécédent. Si une alternative ne satisfait aucune règle, elle est affectée à la catégorie la moins bonne.

Une des forces du modèle est qu'il fournit une justification immédiate de ses préconisations, dans un langage naturel. Il n'est pas demandé au décideur de se plonger dans le mode de fonctionnement du modèle ou de comprendre la signification de paramètres obscurs. Le modèle est très général : il a le même pouvoir d'expression que le modèle précédent, à fonction de valeur croissante et à seuil (Słowiński *et al.*, 2002), à condition toutefois de ne pas borner le nombre de règles composant le modèle. Bien sûr, un modèle comprenant beaucoup de règles peut être très compliqué à comprendre, il se peut donc que cette généralité soit au détriment de la clarté du modèle. L'approche DRSA a été adaptée à de nombreux autres contextes : données manquantes, évaluations floues plutôt que précises, décision de groupe, utilisation d'attributs en plus des critères (Greco *et al.*, 2001b, 2005). L'approche propose aussi des modèles imprécis où une alternative peut être affectée à un intervalle de catégories, de manière à représenter les hésitations ou incertitudes du décideur. Ceci permet de modéliser par exemple des situations où le décideur ne satisfait pas la propriété (critères exhaustifs) ou plus généralement ne respecte pas la dominance. Des modèles basés sur DRSA existent aussi pour les problématiques de rangement et de choix.

Le modèle s'applique particulièrement bien à l'analyse de situations où un grand nombre de décisions passées sont disponibles. Par des modèles de désagrégation, il est alors possible d'inférer des règles et d'observer comment (ou si) ces décisions peuvent s'expliquer à l'aide d'un ensemble de critères donnés.

2.1.3 Modèle de tri à base d'intégrale de Sugeno

Le modèle de tri à base d'intégrale de Sugeno, introduit par Słowiński *et al.* (2002), est paramétré par une fonction μ particulière appelée capacité et des fonctions de valeur partielle $v_j, j \in \mathcal{J}$ (Marichal, 2000). Elle s'inscrit dans la

2.1 Classes de modèles de tri non basées sur le surclassement

logique de la fonction de valeur : elle attribue une valeur dans \mathbb{R} à chaque alternative étant données ses performances, et la catégorie d'affectation des alternatives dépend de la comparaison de leur valeur à des seuils.

Une capacité μ sur un ensemble (ici, de critères) \mathcal{J} est une fonction $\mu : 2^{\mathcal{J}} \mapsto \mathbb{R}_+$ satisfaisant les propriétés suivantes :

bornée inférieurement $\mu(\emptyset) = 0$,

croissante $\forall J_1, J_2 \subseteq \mathcal{J} : J_1 \subseteq J_2 \Rightarrow \mu(J_1) \leq \mu(J_2)$.

Notons que nous suivons la présentation proposée par Bouyssou et Marchant (2007b). D'autres auteurs requièrent parfois que la capacité soit bornée supérieurement à 1 ($\mu(\mathcal{J}) = 1$).

La capacité, dans la façon dont elle est utilisée ici, peut s'interpréter comme suit. Elle indique, pour un sous-ensemble de critère donné, la valeur maximale qu'une alternative peut atteindre en considérant uniquement ces critères. Ainsi, la valeur $\mu(J)$, $J \subseteq \mathcal{J}$, désigne la *capacité* des critères dans J de contribuer à la valeur d'une alternative. La valeur $\mu(J)$ représente une borne supérieure : au mieux, l'alternative atteindra la valeur $\mu(J)$ en considérant les critères dans J , mais elle pourrait être moins bonne, en fonction de ses performances sur ces critères.

Étant donné une capacité μ et des fonctions de valeurs partielles v_j à valeur dans \mathbb{R}_+ , l'intégrale de Sugeno attribue une valeur à une alternative comme suit. Calculons la valeur la plus haute que l'alternative a mérite. L'alternative mérite une valeur v ssi suffisamment de critères sont d'accord avec cette affirmation. Un critère est d'accord avec cette affirmation ssi $v_j(a_j) \geq v$. Notons $J = \{j \in \mathcal{J} \mid v_j(a_j) \geq v\}$ l'ensemble des critères en faveur de l'affirmation selon laquelle a mérite au moins la valeur v . Cette coalition de critères J est suffisante pour attribuer la valeur au moins v à a ssi sa capacité $\mu(J)$ est au moins v . En effet, si la capacité $\mu(J)$ de cet ensemble de critères n'atteint pas v , ces critères ne peuvent pas ensemble attribuer une valeur aussi grande à a . Cela donne la règle suivante : l'alternative a mérite la valeur v ssi $\mu(\{j \in \mathcal{J} \mid v_j(a_j) \geq v\}) \geq v$. Il ne reste plus qu'à choisir la valeur la plus grande que l'alternative mérite :

$$S(a) = \max \{v \mid \mu(\{j \in \mathcal{J} \mid v_j(a_j) \geq v\}) \geq v\},$$

ou, de manière équivalente :

$$S(a) = \max_{J \subseteq \mathcal{J}} \min(\mu(J), \min_{j \in J}(v_j(a_j))).$$

Une fois la valeur des alternatives déterminées, elles peuvent être affectées aux catégories à la façon du modèle à fonction croissante et à seuils : ayant défini des seuils $s^-(C) \in \mathbb{R}$, il suffit de poser $S(a) \geq s^-(C) \Leftrightarrow a \succeq C$.

Notons que Słowiński *et al.* (2002) ont montré que le modèle de tri basé sur l'intégrale de Sugeno est équivalent au modèle de tri à base de règles à condition de restreindre d'une manière précise la façon dont les règles peuvent être formulées.

Par ailleurs, nous verrons plus loin que le modèle basé sur l'intégrale de Sugeno est, de manière surprenante, équivalent à un autre modèle, NCSM sans véto, au fonctionnement apparemment très différent (Słowiński *et al.*, 2002; Bouyssou et Marchant, 2007a,b).

2.2 Classes de modèles proches de ÉLECTRE TRI

Cette section présente le modèle ÉLECTRE TRI et plusieurs variantes de ce modèle. L'essentiel de ce travail s'appuie sur une de ces variantes, MR Sort. Après avoir présenté les concepts communs à toutes ces variantes (section 2.2.1), nous présentons le modèle ÉLECTRE TRI. Le modèle MR-Sort est présenté à la section 2.2.3. Les sections 2.2.4 et 2.2.5 présentent une autre variante, NCSM : il s'agit d'une version simplifiée d'ÉLECTRE TRI qui a été axiomatisée. Cela permet de voir comment ce modèle se compare à d'autres modèles présentés précédemment. De plus, nous verrons que le modèle MR Sort est un cas particulier du modèle NCSM, cela donne donc des indications sur la façon dont MR Sort se compare aux autres modèles exposés.

Nous supposons dans cet exposé que les performances a_j sont numériques et que la préférence correspond au sens croissant des valeurs de performances : $a_j \geq b_j \Rightarrow a_j$ est considéré comme une valeur au moins aussi bonne que b_j sur le critère j .

2.2.1 Concepts communs aux modèles de type ÉLECTRE TRI

Le modèle ÉLECTRE TRI fait partie de la famille ÉLECTRE (Belton et Stewart, 2002; Figueira *et al.*, 2005b), elle même appartenant à une famille de modèles dits de surclassement (Roy, 1985; Roy et Bouyssou, 1993). Ces modèles ont pour point commun qu'ils définissent une relation S indiquant, pour toute paire d'alternatives $(a, b) \in \mathbb{A}$, dans quelle mesure a surclasse b , c'est-à-dire dans quelle mesure a est globalement considérée comme étant au moins aussi bonne que b . Une fois que cette relation est connue, une phase d'exploitation permet de dériver un tri, un rangement, ou un choix. Selon les modèles, la relation de surclassement S peut être une relation floue, donc $S(a, b) \in [0, 1]$, ou une relation binaire, donc $S(a, b) \in \{0, 1\}$. Nous présentons dans cette section des concepts communs à plusieurs modèles de la famille ÉLECTRE. Les

2.2 Classes de modèles proches de ÉLECTRE TRI

annexes A et B décrivent d'autres modèles de la famille ÉLECTRE, ainsi que d'une autre famille proche, PROMÉTHÉE, qui font usage des mêmes concepts.

Pour déterminer si une alternative en surclasse une autre, les modèles de la famille ÉLECTRE prennent en compte deux éléments : la concordance des critères en faveur du surclassement, et la discordance des critères en défaveur.

Un critère se prononce en faveur du surclassement de a sur b ssi la performance de a est suffisamment bonne par rapport à la performance de b . Une relation de concordance partielle K_j sur l'ensemble des alternatives, relation floue ou binaire selon les modèles, indique dans quelle mesure le critère j est d'accord avec le surclassement d'une alternative sur une autre. La concordance générale des critères en faveur du surclassement est déterminée par la somme des concordances partielles K_j pondérée par le poids des critères.

La discordance représente la voix des critères qui se prononcent contre le surclassement. Un critère j est discordant si la performance de a est trop mauvaise par rapport à la performance de b . La discordance est mesurée ici encore par une relation floue ou binaire selon les modèles, notée D_j . Plus la valeur de D_j est élevée, plus le critère s'oppose fermement à un surclassement de a sur b .

Les modèles de surclassement sont en général définis pour des échelles X_j numériques. Les modèles de surclassement utilisent en général les paramètres suivants. Pour chaque critère $j \in \mathcal{J}$, un seuil de préférence p_j indique à partir de quelle différence de performance une alternative est pleinement préférée à une autre, du point de vue du critère j . Le seuil d'indifférence q_j indique la limite en deçà de laquelle une différence de performance n'est pas considérée comme significative. Le seuil de veto v_j représente une valeur considérée comme justifiant une opposition du critère j au surclassement lorsque la différence de performance entre deux alternatives atteint cette valeur. Les valeurs doivent satisfaire $v_j \geq p_j \geq q_j \geq 0$. Le poids $w_j \geq 0$ donne une indication sur l'importance relative du critère j . Enfin, le paramètre λ , appelé seuil de majorité, détermine si les arguments en faveur d'un surclassement sont suffisamment forts. En notant W la somme des poids, le seuil de majorité doit satisfaire $\lambda \in [\frac{W}{2}, W]$. Pour la simplicité de l'exposé, nous supposons que les poids sont normalisés : $\sum_{j \in \mathcal{J}} w_j = W = 1$, donc $\lambda \in [1/2, 1]$.

Pour réduire le risque de confusion avec les notations p_j, q_j , nous désignons dans cette section la performance d'une alternative a sur le critère j par $g_j(a)$ plutôt que par a_j .

Nous donnons ici les définitions permettant de calculer concordance, discordance et surclassement sur lesquelles s'appuient les modèles ÉLECTRE TRI et ses variantes que nous verrons dans la suite de cette section.

La concordance d'un critère j en faveur du surclassement d'une alternative

a sur une alternative b est définie $\forall j \in \mathcal{J}, a, b \in \mathbb{A}$.

$$K_j(a, b) = \begin{cases} 1 & \Leftrightarrow g_j(b) - g_j(a) \leq q_j, \\ \frac{p_j - (g_j(b) - g_j(a))}{p_j - q_j} & \Leftrightarrow q_j < g_j(b) - g_j(a) \leq p_j, \\ 0 & \Leftrightarrow p_j < g_j(b) - g_j(a). \end{cases}$$

Le calcul de la concordance se justifie comme suit. Si l'alternative b n'est pas suffisamment meilleure que a sur le critère j pour dépasser le seuil d'indifférence q_j , le critère j est pleinement en accord avec le surclassement de a sur b . Dans ce cas, la concordance partielle du critère j en faveur du surclassement de a sur b , $K_j(a, b)$, vaut 1. Si au contraire l'alternative b est tellement meilleure que a sur le critère j qu'elle dépasse le seuil de préférence, le critère j n'est aucunement en faveur du surclassement, et $K_j(a, b) = 0$. Entre ces deux extrêmes, la valeur de concordance partielle décroît linéairement de 1 à 0 au fur et à mesure que la différence de performance $g_j(b) - g_j(a)$ s'éloigne de q_j et s'approche de p_j . Roy et Bouyssou (1993, p. 256) notent : « le choix d'une décroissance linéaire peut paraître arbitraire mais tout autre choix (...) serait au moins aussi arbitraire et forcément plus complexe ».

La concordance globale $\Psi(a, b)$ est la somme pondérée des concordances par critère, $\forall a, b \in \mathbb{A}$:

$$\Psi(a, b) = \sum_{j \in \mathcal{J}} w_j K_j(a, b).$$

La discordance du critère j indiquant son opposition au surclassement de a sur b est définie comme suit, $\forall j \in \mathcal{J}, a, b \in \mathbb{A}$.

$$D_j(a, b) = \begin{cases} 1 & \Leftrightarrow v_j < g_j(b) - g_j(a), \\ \frac{(g_j(b) - g_j(a)) - p_j}{v_j - p_j} & \Leftrightarrow p_j < g_j(b) - g_j(a) \leq v_j, \\ 0 & \Leftrightarrow g_j(b) - g_j(a) \leq p_j. \end{cases}$$

Si v_j n'est pas défini, D_j est toujours nul. On a $K_j(a, b) > 0 \Rightarrow D_j(a, b) = 0$, ce qui garantit qu'un même critère ne se prononce pas en même temps en faveur et en défaveur d'un surclassement donné.

Une variante propose de traiter la discordance de façon binaire. Dans ce cas, une relation binaire est définie comme suit, $\forall j \in \mathcal{J}, a, b \in \mathbb{A}$.

$$D_j^{\text{binaire}}(a, b) = \begin{cases} 1 & \Leftrightarrow g_j(b) - g_j(a) > v_j, \\ 0 & \Leftrightarrow g_j(b) - g_j(a) \leq v_j. \end{cases}$$

Ici encore, si v_j n'est pas défini, D_j^{binaire} est toujours nul. Notons que dans le cas binaire, ne pas définir v_j est équivalent à choisir une valeur suffisamment

2.2 Classes de modèles proches de ÉLECTRE TRI

grande pour ne jamais activer le véto. Choisir $v_j \geq g_j(b) - g_j(a), \forall a, b \in \mathbb{A}$ (si une telle valeur existe) permet de s'en assurer. Observons également que si $p_j = v_j$, alors $D_j = D_j^{\text{binaire}}$. Enfin, on a bien $K_j(a, b) > 0 \Rightarrow D_j^{\text{binaire}}(a, b) = 0$.

La connaissance des valeurs de concordance et de discordance permet de calculer la relation de surclassement $S, \forall a, b \in \mathbb{A}$.

$$S(a, b) = \begin{cases} 0 & \Leftrightarrow \exists j \in \mathcal{J} \mid D_j(a, b) = 1, \\ \Psi(a, b) \prod_{\{j \mid D_j(a, b) > \Psi(a, b)\}} \frac{1 - D_j(a, b)}{1 - \Psi(a, b)} & \text{sinon.} \end{cases} \quad (\text{surclassement})$$

Les modèles de la famille ÉLECTRE sont habituellement décrits sous l'hypothèse implicite qu'aucun critère de poids nul n'est inclus dans l'ensemble des critères considérés. Sous cette hypothèse, la définition de S ci-dessus peut être simplifiée :

$$S^{\text{simplifié}}(a, b) = \Psi(a, b) \prod_{\{j \mid D_j(a, b) > \Psi(a, b)\}} \frac{1 - D_j(a, b)}{1 - \Psi(a, b)}. \quad (\text{surclassement simplifié})$$

Observons que $D_j(a, b) = 1 \Rightarrow K_j(a, b) = 0$, donc, sous l'hypothèse qu'aucun véto n'est associé à un critère de poids nul, on a : $[\exists j \in \mathcal{J} \mid D_j(a, b) = 1] \Rightarrow \Psi(a, b) < 1$. Cela entraîne l'équivalence de S et $S^{\text{simplifié}}$ sous cette hypothèse.

2.2.2 Modèle ÉLECTRE TRI

Le modèle ÉLECTRE TRI (Mousseau *et al.*, 2000) est un modèle de tri imprécis : une alternative peut être affectée à un intervalle de catégories dans \mathcal{C} , et pas nécessairement à une catégorie unique. Il utilise comme paramètres les seuils de préférence p_j , d'indifférence q_j , de véto v_j , les poids w_j , pour chaque critère j , et un seuil de majorité λ . Il utilise en outre un ensemble L de profils séparateurs de catégories. Chaque catégorie $C \in \mathcal{C}$ est associée à un profil limite inférieur, noté $l^-(C)$, et à un profil limite supérieur noté $l^+(C)$. Le profil supérieur d'une catégorie coïncide avec le profil inférieur de la catégorie qui lui est immédiatement préférée, comme l'indique la figure 2.2. Un profil, à la manière d'une alternative, est défini par une valeur de performance sur chaque critère $j \in \mathcal{J}$, performance choisie dans X_j . Le modèle décide de l'affectation des alternatives d'après la façon dont elles se comparent aux profils. Nous supposons que les performances des profils sont ordonnées de façon compatibles aux catégories auxquelles les profils sont associés : $\forall j \in \mathcal{J}$,

$$C \preceq C' \Rightarrow l_j^-(C) \leq l_j^-(C'). \quad (2.1)$$

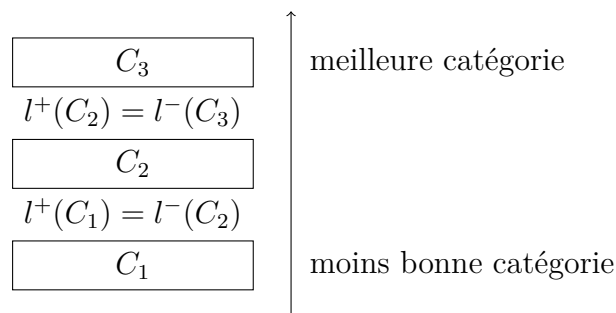


FIGURE 2.2 – Trois catégories et les profils qui les séparent.

Le modèle ÉLECTRE TRI utilise les relations floues K_j , Ψ , D_j , S définies ci-dessus et introduit également une relation S^λ , correspondant binaire de la relation floue S , définie comme suit, $\forall a, b \in \mathbb{A}$.

$$S^\lambda(a, b) = \begin{cases} 1 & \Leftrightarrow S(a, b) \geq \lambda \\ 0 & \Leftrightarrow S(a, b) < \lambda. \end{cases}$$

C'est cette relation qui sera utilisée pour comparer les alternatives aux profils. Notons qu'il est possible d'avoir non $S^\lambda(a, b)$ et non $S^\lambda(b, a)$, autrement dit la relation binaire S^λ n'est pas complète. Cela arrive par exemple si a est considérée comme pleinement meilleure que b (donc $K_j(a, b) = 1$) sur une moitié de critères, et b est considérée comme pleinement meilleure que a sur l'autre moitié des critères, avec des poids égaux, et $\lambda > 0,5$. Dans une telle situation, les arguments conduisant à trouver a meilleure que b sont considérés comme globalement insuffisants, tout comme les arguments conduisant à la conclusion inverse. Une telle situation conduit à deux conclusions possibles. Un regard optimiste suggère de considérer l'alternative a comme aussi bonne que b , car les arguments contraires sont insuffisants. Une position plus exigeante amènerait à demander des arguments positifs en faveur d'un surclassement avant de le considérer comme valide, et donc à considérer que a n'est pas aussi bonne que b .

Le modèle propose deux méthodes d'affectation des alternatives aux catégories, la méthode optimiste et la méthode pessimiste. La méthode pessimiste affecte chaque alternative à une catégorie moins bonne ou égale à la catégorie à laquelle la procédure optimiste affecte l'alternative, d'où leur nom.

La méthode pessimiste définit la relation binaire $\succeq^{\text{pessimiste}}$ sur l'ensemble $\mathbb{A} \times \mathcal{C}$, qui indique, pour une paire (a, C) , si l'alternative a mérite d'être affectée à la catégorie C . Pour la méthode pessimiste, une alternative mérite d'être affectée à une catégorie ssi elle est au moins aussi bonne que la limite inférieure

2.2 Classes de modèles proches de ÉLECTRE TRI

de cette catégorie : $\forall a \in \mathbb{A}, C \in \mathcal{C}, C \neq C_1,$

$$a \succeq^{\text{pessimiste}} C \Leftrightarrow S^\lambda(a, l^-(C)).$$

On définit également $a \succeq^{\text{pessimiste}} C_1 = 1$. En effet, la moins bonne catégorie, C_1 , n'a pas de profil inférieur. Il n'est pas nécessaire, car toute alternative mérite au moins d'accéder à la plus mauvaise catégorie. L'alternative est finalement affectée à la meilleure catégorie parmi celles auxquelles elle mérite d'accéder. Intuitivement, en se référant à la figure 2.2, la procédure pessimiste parcourt donc les catégories du bas vers le haut, en regardant à chaque frontière si l'alternative peut atteindre la catégorie suivante.

La procédure optimiste fonctionne de manière similaire, mais accorde plus facilement l'accès à une catégorie. Selon la procédure optimiste, une alternative a mérite d'accéder à une catégorie C ssi elle est au moins aussi bonne que la limite inférieure de la catégorie, ou qu'aucune des deux affirmations de préférence ne tient : la limite inférieure de la catégorie n'est pas au moins aussi bonne que l'alternative, et l'alternative n'est pas au moins aussi bonne que la catégorie. $\forall a \in \mathbb{A}, C \in \mathcal{C}, C \neq C_1 :$

$$a \succeq^{\text{optimiste}} C \Leftrightarrow S^\lambda(a, l^-(C)) \vee (\neg S^\lambda(a, l^-(C)) \wedge \neg S^\lambda(l^-(C), a)).$$

De même que dans le cas pessimiste, on définit également $a \succeq^{\text{optimiste}} C_1 = 1$.

Une particularité d'ÉLECTRE TRI

Observons une caractéristique d'ÉLECTRE TRI qui peut être considérée comme une faiblesse de ce mode de raisonnement (Bouyssou *et al.*, 2006b, p. 381). Il peut arriver qu'une alternative a soit classée dans une meilleure catégorie qu'une alternative b alors même qu'en comparant directement a à b (ce que le modèle ÉLECTRE TRI ne fait pas), on aurait observé une préférence inverse. Considérons par exemple le problème suivant à deux catégories et trois critères, en utilisant pour chaque critère les entiers $\{-2, -1, 0, 1, 2\}$, avec un ordre de préférence correspondant à l'ordre classique \geq sur les entiers. Ces entiers peuvent avoir pour sémantique $\{\text{nul, mauvais, moyen, bon, excellent}\}$. Les seuils de véto ne sont pas utilisés, et $p_j = q_j = 0$. Le profil séparateur $l^-(C_2) = l^+(C_1)$ égale $\langle \text{moyen, moyen, moyen} \rangle = \langle 0, 0, 0 \rangle$, les critères ont des poids égaux $w_j = 1/3$, et le seuil de majorité λ vaut $1/2$. Considérons deux alternatives $a = \langle 1, 1, -2 \rangle = \langle \text{bon, bon, nul} \rangle$ et $b = \langle 2, -1, -1 \rangle = \langle \text{excellent, mauvais, mauvais} \rangle$. On voit que $a \in C_2, b \in C_1$, alors que $S(b, a) = 2/3$, donc $S(b, a) \geq \lambda$, donc b serait meilleur que a si on les comparait directement en utilisant la même logique de surclassement.

Ceci montre que le mode de raisonnement adopté par ÉLECTRE TRI peut parfois entrer en contradiction avec d'autres modes de raisonnements. Dans

l'exemple ci-dessus, la justification du placement de l'alternative a dans la meilleure catégorie est qu'elle est « au moins moyenne » sur deux critères, alors que b n'est « au moins moyenne » que sur un critère. Un tel raisonnement n'exclut pas, l'exemple le montre, que b puisse toutefois être meilleure que a en comparaison directe. Il est donc important de valider avec le décideur l'acceptation des règles induites par le choix du modèle de préférence, y compris dans les cas où ces règles en contredisent d'autres qui semblent également acceptables. Cette remarque est valable au-delà de l'exemple particulier donné ici pour ÉLECTRE TRI, car tout modèle de préférence précis (ou insuffisamment imprécis) pourrait manquer de modéliser les différents modes de raisonnement qui peuvent conduire à des conclusions contradictoires. Ceci illustre les réflexions proposées à la section 1.3.

2.2.3 Modèle MR Sort

Le modèle ÉLECTRE TRI peut être simplifié de la façon suivante, qui permet d'obtenir un modèle plus simple à analyser et à expliquer en gardant les principes du modèle. Le modèle est principalement justifié par l'existence d'une axiomatisation d'un modèle similaire, que nous présentons juste après celui-ci (Sections 2.2.4 et 2.2.5). Nous nommons ce modèle simplifié MR Sort (Majority Rule Sort), comme proposé par Leroy *et al.* (2011). Il s'agit du modèle principal utilisé dans ce travail. Cette section se conclut par une brève discussion concernant les forces et les faiblesses de ce modèle.

La simplification consiste à négliger les seuils de préférence et d'indifférence en posant $p_j = q_j = 0$, à considérer uniquement des vétos binaires, et uniquement la règle d'affectation pessimiste. L'utilisation de la règle pessimiste seule en fait un modèle précis, c'est-à-dire un modèle qui affecte chaque alternative à une et une seule catégorie.

Échelles en termes d'intervalles

Négligeons les vétos dans un premier temps, et observons les conséquences de ces simplifications. La règle d'affectation d'une alternative à une catégorie devient :

$$a \succeq C \Leftrightarrow \sum_{j \in \mathcal{J} | a_j \geq l_j^-(C)} w_j \geq \lambda.$$

Cela simplifie l'interprétation du rôle des échelles X_j : chaque échelle X_j peut être vue comme un ensemble d'intervalles emboîtés relatifs chacun à une catégorie donnée. La figure 2.3 montre un exemple d'échelle X_j . L'intervalle de valeurs supérieures ou égales à $l_j^-(C_3)$, noté $\mathcal{L}_j(C_3)$ sur la figure, comprend

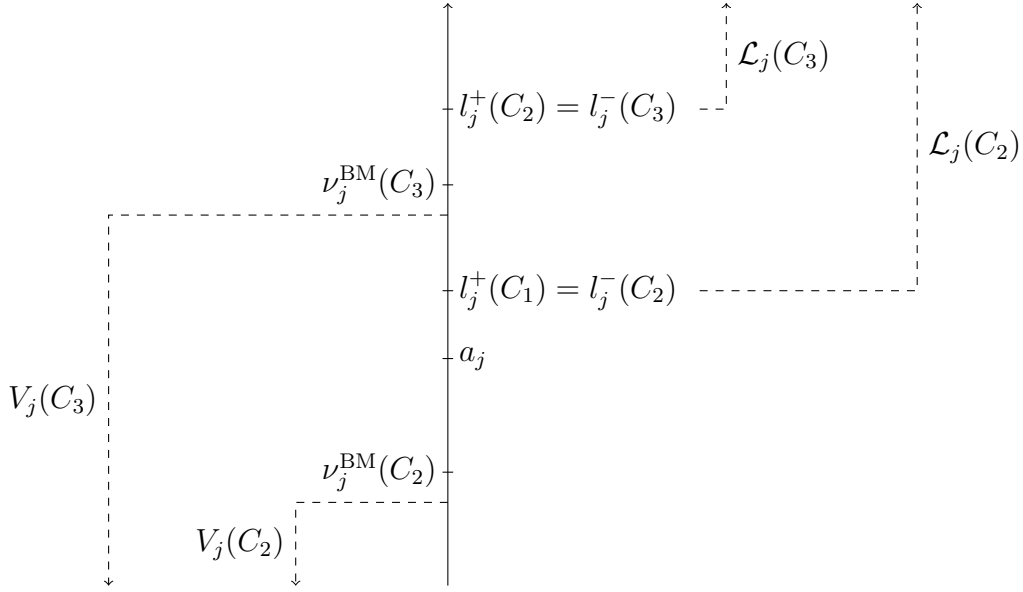


FIGURE 2.3 – Une échelle d'un critère avec des profils et des vétos

toutes les valeurs de performances favorables à l'affectation de a à C_3 : le poids du critère j n'est accordé en faveur d'une affectation de a à C_3 que lorsque la performance de a sur le critère j est au moins $l_j^-(C_3)$. L'intervalle de valeurs supérieures ou égales à $l_j^-(C_2)$ est favorable à l'affectation en C_2 : le poids du critère compte dans la coalition en faveur de $a \succeq C_2$ ssi la performance de a sur le critère j est dans cet intervalle. Cet ensemble de valeurs est noté $\mathcal{L}_j(C_2)$ sur la figure, et on voit que $\mathcal{L}_j(C_3) \subset \mathcal{L}_j(C_2)$. Enfin, l'intervalle complet X_j est favorable à l'affectation en C_1 : les critères sont tous en faveur de l'affirmation $a \succeq C_1$. Ainsi, fixer des valeurs pour les performances des profils sur le critère j détermine une découpe de l'échelle X_j en intervalles emboîtés.

Définition alternative du veto

Dans cette version utilisant des vétos binaires, il est plus intuitif de considérer une autre définition du seuil de veto. Posons $\forall j \in \mathcal{J}, C \in \mathcal{C} \setminus C_1$:

$$\nu_j^{\text{BM}}(C) = l_j^-(C) - v_j.$$

ÉLECTRE TRI exige $v_j \geq 0$, donc exigeons de manière équivalente $\nu_j^{\text{BM}}(C) \leq l_j^-(C)$.

En appliquant la définition de la discordance binaire à la comparaison entre

une alternative et un profil, on obtient, $\forall a \in \mathbb{A}, C \in \mathcal{C}$:

$$D_j^{\text{binaire}}(a, l^-(C)) = \begin{cases} 1 & \Leftrightarrow l^-(C) - a_j > v_j \Leftrightarrow a_j < \nu_j^{\text{BM}}(C), \\ 0 & \text{sinon.} \end{cases}$$

De manière équivalente à la définition classique d'ÉLECTRE TRI, ce véto intervient lorsque l'alternative a une performance trop basse sur le critère j . Cette notation facilite l'interprétation : lorsque l'alternative a a une performance sur un critère j inférieure à $\nu_j^{\text{BM}}(C)$, il est interdit à cette alternative de surclasser $l^-(C)$, la limite inférieure de C , donc l'alternative a se voit interdire l'accès à la catégorie C . Choisir $\nu_j^{\text{BM}}(C) \leq a_j, \forall a \in \mathbb{A}$ permet de ne jamais activer le véto, si une telle valeur existe. Même dans le cas où les performances a_j n'admettent pas de minimum, il est toujours possible de décider de ne pas utiliser le véto, pour un critère j et une catégorie C donnés. Nous dirons dans ce cas que la valeur en question n'est pas définie. Notons que $C \preceq C' \Rightarrow \nu_j^{\text{BM}}(C) \leq \nu_j^{\text{BM}}(C')$, étant donné (2.1) page 45.

Ainsi, fixer les seuils de véto détermine une autre découpe de l'échelle X_j , similaire en ce qu'elle crée également des intervalles emboîtés qu'on peut associer aux catégories, mais avec une sémantique différente. En se reportant à nouveau à la figure 2.3, si une alternative a une performance a_j située dans la zone en-dessous de $\nu_j^{\text{BM}}(C_3)$ désignée par $V_j(C_3)$, elle ne pourra pas être affectée en catégorie C_3 . Si elle a une performance située dans la zone en-dessous de $\nu_j^{\text{BM}}(C_2)$ désignée par $V_j(C_2)$, elle ne pourra pas atteindre la catégorie C_2 ou une catégorie meilleure que C_2 . On a bien $V_j(C_2) \subseteq V_j(C_3)$.

La règle d'affectation d'une alternative à une catégorie lorsque les vétos jouent est la suivante :

$$a \succeq C \Leftrightarrow \left[\sum_{j \in \mathcal{J} | a_j \geq l_j^-(C)} w_j \geq \lambda \right] \wedge [a_j \geq \nu_j^{\text{BM}}(C), \forall j \in \mathcal{J}].$$

(affectation MR Sort)

Le lecteur pourra vérifier que cette règle est bien la règle d'affectation d'ÉLECTRE TRI dans le cadre des simplifications apportées par MR Sort et en utilisant le mode de calcul de surclassement général indiqué précédemment (voir (surclassement) page 45).

Notons pour terminer cette présentation que le nom « Majority Rule Sort » n'est pas parfaitement approprié lorsque des vétos sont autorisés. Les auteurs à l'origine du nom (Leroy *et al.*, 2011) n'utilisent que la variante sans véto. Nous conservons cependant ce nom même dans le cas où les vétos sont utilisés par commodité. Il faudrait idéalement appeler le modèle « MR Sort avec vétos » dans ce cas.

Discussion du modèle

Ces transitions brusques d'un intervalle favorable à une catégorie à l'autre rendent la sémantique des échelles et le fonctionnement du modèle plus clairs, mais elles peuvent également être vues comme une faiblesse du modèle. Il peut être insatisfaisant que la modification d'une performance d'une alternative, faible mais suffisante pour franchir une frontière tracée par un profil ou un seuil de véto, entraîne le basculement total du poids du critère considéré, voire le refus catégorique d'affecter l'alternative à une catégorie donnée. Il serait bon d'évaluer empiriquement si ce comportement est acceptable en situation concrète aux yeux des décideurs. De tels tests pourraient s'appuyer sur l'axiomatisation d'un modèle proche, NCSM. Pour ce qui concerne les seuils de véto, il paraît plausible que ce comportement puisse être considéré comme raisonnable lorsque le véto concerne une échelle discrète. L'évaluation d'étudiants est souvent envisagée d'une manière similaire : une note insuffisante sur une matière peut suffire à faire échouer l'étudiant, indépendamment de ses notes sur les autres matières.

Nous pensons que, dans l'état actuel des connaissances en la matière, la simplicité plus grande du modèle et son intérêt théorique dû à sa proximité d'un modèle axiomatisé compensent les éventuelles faiblesses dont il pourrait être accusé. Surtout, nous pensons que l'approche consistant à modéliser un comportement de véto est digne d'intérêt et postulons que des recherches expérimentales, malheureusement trop rares, fourniraient des arguments en sa faveur.

2.2.4 Modèle NCSM sans véto

Bouyssou et Marchant (2007a,b) ont axiomatisé un modèle similaire à ÉLECTRE TRI qu'ils ont baptisé NCSM. Les preuves des affirmations concernant le modèle NCSM exposées ici et dans la section suivante sont disponibles dans ces deux articles. Dans le cas sans véto, des études précédentes avaient proposé une analyse similaire (Greco *et al.*, 2001a ; Słowiński *et al.*, 2002).

Nous présentons d'abord le modèle n'utilisant pas de véto. Le modèle fait intervenir des ensembles emboîtés $\mathcal{L}_j(C) \subseteq X_j, \forall j \in \mathcal{J}, C \in \mathcal{C} \setminus C_1$ et un ensemble de coalitions suffisantes $F(C) \subseteq 2^{\mathcal{J}}$ associé à chaque catégorie $C \in \mathcal{C}$. Les ensembles $\mathcal{L}_j(C)$ doivent satisfaire $C \supseteq C' \Rightarrow \mathcal{L}_j(C) \subseteq \mathcal{L}_j(C')$. L'ensemble $\mathcal{L}_j(C)$ contient les valeurs de a_j qui amènent le critère j à autoriser a à accéder à C , ainsi qu'indiqué en section 2.2.3.

Les coalitions suffisantes $F(C)$ associées à une catégorie C indiquent quels ensembles de critères sont suffisants pour autoriser l'accès d'une alternative à C . Tous les ensembles de coalitions suffisantes $F(C)$ satisfont la propriété sui-

vante : $J \in F(C) \Rightarrow J' \in F(C), \forall J \subseteq J' \subseteq \mathcal{J}$. Autrement dit, si un ensemble de critères J est considéré comme suffisant pour permettre à a d'accéder à C , alors un surensemble de critères J' doit aussi être suffisant. Les ensembles de coalitions suffisantes sont également soumis à la condition suivante : si un ensemble de critères peut octroyer l'accès à une catégorie C , il doit pouvoir également octroyer l'accès à une catégorie moins bonne C' . Ainsi, les ensembles de coalitions doivent satisfaire $C \succeq C' \Rightarrow F(C) \subseteq F(C')$.

Les affectations sont calculées comme suit, $\forall a \in \mathbb{A}, C \in \mathcal{C}$:

$$a \succeq C \Leftrightarrow \{j \in \mathcal{J} \mid a_j \in \mathcal{L}_j(C)\} \in F^C.$$

Notons également que les résultats axiomatiques indiqués pour les modèles NCSM supposent que tous les critères sont influents. Un critère j est influent ssi $\exists x_j, y_j \in X_j, a_{-j} \in X_{-j}, C, C' \in \mathcal{C}, C \neq C' \mid (x_j, a_{-j}) \in C, (y_j, a_{-j}) \in C'$.

L'axiomatisation de ce modèle s'exprime très simplement. Définissons la propriété (2-graded- j forte) : $\forall x_j, y_j, z_j \in X_j, C, C' \in \mathcal{C}, C' \preceq C, a_{-j}, b_{-j} \in X_{-j}$,

$$\left. \begin{array}{l} (z_j, a_{-j}) \prec C \\ \text{et} \\ (x_j, a_{-j}) \succeq C \\ \text{et} \\ (y_j, b_{-j}) \succeq C' \end{array} \right\} \Rightarrow (x_j, b_{-j}) \succeq C' \quad \text{(2-graded-}j \text{ forte)}$$

Si les préférences du décideur satisfont (2-graded- j forte), pour tout $j \in \mathcal{J}$, alors il existe des ensembles $\mathcal{L}_j(C)$ et des ensembles de coalitions suffisantes $F(C)$ représentant ces préférences selon le modèle NCSM sans véto. La condition est également nécessaire. Cette condition peut se comprendre de la manière suivante. La première prémisse indique que les valeurs a_{-j} ne sont pas suffisantes pour garantir l'accès d'une alternative à C . Par contre, lorsque z_j est remplacé par x_j , l'alternative devient suffisamment bonne pour pouvoir y accéder. Observons que les variables sont définies telles que $C' \preceq C$. Dans ces conditions, considérant une alternative $(y_j, b_{-j}) \succeq C'$, cette propriété demande que le remplacement de y_j par x_j ne nuise pas à l'accès de l'alternative à C' .

La figure 2.4 offre un autre point de vue pour expliquer cette propriété. Positionnons les valeurs sur l'axe X_j de manière à satisfaire les prémisses de la propriété. Il faut que x_j soit meilleur que z_j pour respecter la dominance. La valeur y_j peut être n'importe où sur l'axe. Dans cette situation, l'implication tient dans le modèle NCSM sans véto pour la raison suivante. La frontière de C sur X_j est nécessairement entre x_j et z_j étant donné les deux premières prémisses, et la frontière de C' ne peut être au-dessus de celle de C car $C' \preceq C$. Ceci garantit que x_j est dans l'ensemble de valeurs en faveur de C' , donc que

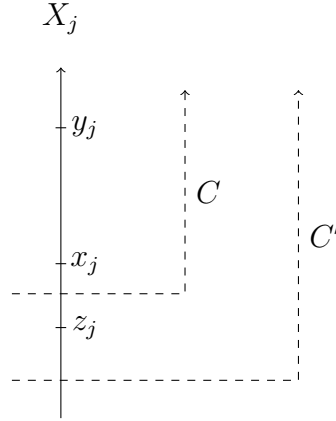


FIGURE 2.4 – Une échelle d'un critère illustrant la condition 2-graded- j forte.

remplacer une évaluation y_j par x_j pour une alternative affectée au moins à C' , même si x_j est inférieure à y_j comme le propose cette figure, ne fournit pas un argument pour dégrader l'alternative en-dessous de C' .

Cette explication fournit aussi un exemple de situation où la propriété n'est pas respectée par un modèle qui respecte la dominance, ce qui montre que le modèle NCSM sans véto n'est pas aussi général que le modèle à fonction croissante et à seuils. En effet, dans un modèle à fonction d'agrégation plus général, utilisant par exemple une fonction d'agrégation additive, il est parfaitement possible que remplacer l'évaluation y_j par une évaluation x_j inférieure interdise à l'alternative l'accès à C' .

Pour montrer plus précisément le lien entre le modèle général et celui-ci, la propriété (2-graded- j) définie elle aussi par Bouyssou et Marchant (2007a,b) peut être utile : $\forall x_j, y_j, z_j \in X_j, C, C' \in \mathcal{C}, C' \preceq C, a_{-j}, b_{-j} \in X_{-j}$,

$$\left. \begin{array}{l} (z_j, a_{-j}) \prec C \\ \text{et} \\ (x_j, a_{-j}) \succeq C \\ \text{et} \\ (y_j, a_{-j}) \succeq C \\ \text{et} \\ (y_j, b_{-j}) \succeq C' \end{array} \right\} \Rightarrow (x_j, b_{-j}) \succeq C' \quad (2\text{-graded-}j)$$

La condition (2-graded- j) doit être ajoutée à (linear- j) pour passer du modèle à fonction croissante et à seuil à celui-ci. Le lien entre les deux modèles est établi par :

$$(\text{linear-}j) \wedge (2\text{-graded-}j) \Leftrightarrow (2\text{-graded-}j \text{ forte}).$$

La classe de modèles MR Sort sans véto est incluse dans la classe NCSM sans véto. En effet, il est toujours possible de représenter un modèle MR Sort sans véto comme un modèle NCSM sans véto de la façon suivante. Les coalitions sont déterminées indépendamment de la catégorie : $\forall C \in \mathcal{C}$,

$$F(C) = \{J \in 2^{\mathcal{J}} \mid \sum_{j \in J} w_j \geq \lambda\}.$$

Les ensembles $\mathcal{L}_j(C)$ sont définis comme suit, $\forall j \in \mathcal{J}, C \in \mathcal{C} \setminus C_1, a_j \in X_j$:

$$a_j \in \mathcal{L}_j(C) \Leftrightarrow a_j \geq l_j^-(C).$$

Le modèle basé sur l'intégrale de Sugeno à seuils est équivalent au modèle NCSM sans véto, c'est-à-dire que les préférences du décideur peuvent être représentées dans un des modèles ssi elles peuvent être représentées dans l'autre. Ce résultat est dû à Słowiński *et al.* (2002).

2.2.5 Modèle NCSM avec véto

Le modèle NCSM avec véto admet comme paramètre supplémentaire un seuil de véto ν_j^{BM} , défini sur chaque critère. Il est utilisé comme indiqué à la section 2.2.3. Les alternatives sont donc comparées aux catégories de la façon suivante :

$$a \succeq C \Leftrightarrow [\{j \in \mathcal{J} \mid a_j \succeq_j l_j^-(C)\} \in F^C] \wedge [\forall j \in \mathcal{J} : a_j \geq \nu_j^{\text{BM}}(C)].$$

Nous supposons ici, par simplicité, qu'il existe une représentation numérique permettant de représenter les vétos et les performances comme des nombres. Le modèle est défini plus généralement à l'aide d'ensembles emboîtés V_j .

Définissons la propriété (3v-graded- j) : $\forall x_j, y_j, z_j \in X_j, C, C' \in \mathcal{C}, C' \trianglelefteq C, a_{-j}, b_{-j} \in X_{-j}$:

$$\left. \begin{array}{l} (z_j, a_{-j}) \prec C \\ \text{et} \\ (x_j, a_{-j}) \succeq C \\ \text{et} \\ (y_j, a_{-j}) \succeq C \\ \text{et} \\ (z_j, c_{-j}) \succeq C \\ \text{et} \\ (y_j, b_{-j}) \succeq C' \end{array} \right\} \Rightarrow (x_j, b_{-j}) \succeq C'. \quad (3\text{v-graded-}j)$$

Il existe une représentation des préférences du décideur dans le modèle NCSM avec veto ssi elles satisfont (linear- j) et (3v-graded- j).

Notons que le modèle MR Sort avec veto est un cas particulier de NCSM avec veto.

L'annexe C contient une note concernant les vetos associés aux critères de poids nul dans les modèles NCSM et MR Sort.

2.3 Méthodes relatives à la problématique de tri

Les sections 1.6, 1.7 et 1.8 ont présenté de façon générale les approches de la robustesse, de la désagrégation et le traitement des inconsistances. Nous donnons dans cette section des détails concernant les travaux relatifs à la désagrégation au sens large (incluant les préoccupations de robustesse et de traitement d'inconsistance) de modèles de tri, et particulièrement sur les travaux considérant des variantes proches de ÉLECTRE TRI.

2.3.1 Désagrégation par la méthode UTADIS

La méthode UTADIS (Devaud *et al.*, 1980) est une méthode qui infère un modèle UTADIS à partir d'exemples d'affectation précis donnés par un décideur. Pour ce faire, un PM est défini. Le programme utilise comme variables les paramètres du modèle de préférence à trouver, et représente les exemples d'affectation comme des contraintes.

Pour rappel, le modèle UTADIS (présenté à la section 2.1.1) requiert la définition de fonctions de valeur partielles $v_j, j \in \mathcal{J}$, et de seuils $s^-(C), C \in \mathcal{C} \setminus \mathcal{C}_1$.

Notons $X \subseteq \mathbb{A}$ l'ensemble des alternatives utilisées comme exemples par le décideur. Notons $(x \rightarrow C)$ un exemple indiquant qu'une alternative $x \in X$ est affectée par le décideur à la catégorie $C \in \mathcal{C}$.

La méthode utilise comme fonctions de valeurs partielles v_j des fonctions linéaires par morceaux : l'espace de performances d'un critère X_j est découpé en un ensemble fini de n_j morceaux, notés $g_j^0 = \underline{g}_j, \dots, g_j^k, \dots, g_j^{n_j} = \overline{g}_j$. Les performances minimales et maximales atteintes par l'ensemble des alternatives utilisées en exemples sont notées $\underline{g}_j = \min_{x \in X} x_j$ et $\overline{g}_j = \max_{x \in X} x_j$. Le nombre de morceaux n_j et les valeurs \underline{g}_j sont fixés a priori.

Il faut déterminer les valeurs de la fonction v_j en les points g_j^k , ce qui détermine la fonction v_j . Les valeurs $v_j(g_j^k), \forall j \in \mathcal{J}, 0 \leq k \leq n_j$, sont donc des variables du Programme Linéaire (PL).

Une fois ces valeurs déterminées, la valeur d'une alternative sur le critère j ,

$v_j(a_j)$, peut être calculée. Cette valeur vaut, avec $g_j^k \preceq_j a_j \preceq_j g_j^{k+1}$:

$$v_j(a_j) = v_j(g_j^k) + \frac{v_j(g_j^{k+1}) - v_j(g_j^k)}{g_j^{k+1} - g_j^k}(a_j - g_j^k).$$

Les seuils $s^-(C)$, $C \in \mathcal{C} \setminus C_1$ sont également des variables du problème.

Pour que les seuils soient ordonnées correctement, la contrainte suivante est utilisée, $\forall C_h \in \mathcal{C} \setminus C_1, C_{nc}$:

$$s^-(C_h) < s^-(C_{h+1}).$$

De même, les valeurs des échelons $v_j(g_j^k)$ doivent être ordonnés correctement pour garantir que les fonctions de valeurs soient monotones croissantes : $\forall j \in \mathcal{J}, 0 \leq k < n_j$,

$$v_j(g_j^k) < v_j(g_j^{k+1}).$$

La somme de toutes les valeurs $v_j(\bar{g}_j)$ et l'origine des fonctions de valeur peuvent être fixées librement :

$$\sum_{j \in \mathcal{J}} v_j(\bar{g}_j) = 1,$$

$\forall j \in \mathcal{J}$:

$$v_j(\underline{g}_j) = 0.$$

Les contraintes suivantes tentent de respecter les exemples d'affectation du décideur autant que possible. $\forall (x \rightarrow C_h)$:

$$\begin{aligned} v(x) - \sigma^+(x) &< s^-(C_{h+1}), \\ v(x) + \sigma^-(x) &\geq s^-(C_h). \end{aligned}$$

La fonction de valeur $v(x)$ vaut $\sum_{j \in \mathcal{J}} v_j(x_j)$. Les valeurs $\sigma^+(x), \forall x \in X$ et $\sigma^-(x), \forall x \in X$ sont des variables du PL, définies non négatives. Elles représentent respectivement l'erreur de sur-estimation et l'erreur de sous-estimation de la valeur de x . Idéalement, ces erreurs devraient valoir zéro, ce qui indiquerait que le modèle trouvé satisfait parfaitement les exemples du décideur. L'introduction de ces variables permet de trouver un modèle même quand les exemples ne sont pas tous représentables avec un modèle UTADIS et des fonctions de valeurs linéaires par morceaux.

La fonction objectif consiste à minimiser la somme des erreurs :

$$\min_{x \in X} \sigma^+(x) + \sigma^-(x).$$

Des variantes de cette méthode ont été proposées, appelées UTADIS I, II, III (Zopounidis et Doumpos, 1997, 2001 ; Doumpos et Zopounidis, 2002). UTADIS I maximise la séparation entre les valeurs des alternatives et les seuils. UTADIS II minimise le nombre d'erreurs, ce qui transforme le PM en Programme Linéaire en Nombres Entiers (PLNE). UTADIS III combine ces deux objectifs. Zopounidis et Doumpos (2000) ont implémenté ces variantes dans un système d'aide à la décision appelé PREFDIS.

2.3.2 Approche robuste par sous-ensemble de paramètres

Greco *et al.* (2010) ont proposé une approche robuste basée sur un modèle UTADIS, donc un modèle de tri à fonction croissante et à seuils utilisant une fonction d'agrégation additive (voir Section 2.1.1). Nous exposons ici le principe d'une variante de cette méthode. Nous prenons la liberté de ne pas suivre l'exposé original, plus complexe que nécessaire pour en comprendre les principes qui nous intéressent ici. Nous choisissons aussi de présenter cette variante selon deux axes, le premier en utilisant principalement les concepts de la robustesse exposés à la section 1.6, le second en passant par une classe de modèles imprécis évoqués à la section 1.5. Ce second axe est plus proche de l'esprit de la méthode proposée par les auteurs (Greco *et al.*, 2010). Présenter ces deux axes d'analyse permet d'illustrer que ces deux approches peuvent être liées, un terrain de recherche qui nous semble intéressant mais à notre connaissance inexploré.

Notons comme d'habitude M la classe de modèles choisie, ici, le modèle UTADIS. Notons Ω^* l'ensemble des paramètres possibles, contenant ici l'ensemble des fonctions de valeurs v possibles et l'ensemble des valeurs de seuils $s^-(C)$, $C \in \mathcal{C} \setminus C_1$ possibles. Appeler Ω^* un ensemble de paramètres d'un modèle est ici un abus de langage étant donné que Ω^* contient des fonctions, mais encore une fois nous passons outre cette difficulté technique pour nous concentrer sur l'exposé des concepts. L'ensemble des fonctions de valeurs v possibles contient les fonctions admissibles dans le modèle et seulement celles-là, c'est-à-dire que ces fonctions doivent pouvoir s'exprimer comme $V(v_j(a_j), j \in \mathcal{J})$, avec des fonctions de valeurs partielles v_j croissantes, comme indiqué à la section 2.1.1.

La méthode a ceci d'original qu'elle traite distinctement deux types de paramètres : les fonctions de valeur et les seuils. Notons \mathcal{V}^* l'ensemble des fonctions de valeurs possibles, c'est-à-dire le produit cartésien de l'ensemble des fonctions de valeur partielles possibles. Notons \mathcal{S}^* l'ensemble des valeurs de seuils possibles. On a $\Omega^* = \mathcal{V}^* \times \mathcal{S}^*$. Choisir une fonction de valeur $v \in \mathcal{V}^*$ et des seuils $S \in \mathcal{S}^*$ détermine complètement le modèle, et permet donc de définir une fonction de tri $f_{v,S} = f_\omega$, avec $\omega = (v, S)$, $\omega \in \Omega^*$. Cette fonction affecte les

alternatives $a \in \mathbb{A}$ à la manière dictée par le modèle de tri à fonction croissante et à seuils, donc en comparant la valeur de l'alternative aux seuils des catégories.

Considérons maintenant un ensemble E d'affectations imprécises fournies par un décideur. L'explication donnée ici suppose que cet ensemble d'exemples est cohérent, donc peut être représenté par une fonction de tri dans M . L'ensemble contient des exemples notés $(x \rightarrow [\underline{C}_x, \overline{C}_x])$. Notons $X \subseteq \mathbb{A}$ l'ensemble des alternatives utilisées en exemples. Nous dirons qu'une fonction de tri $f_{v,S}$ satisfait les exemples, suivant les principes classiques de la désagrégation, ssi $\forall x \in X$:

$$f_{v,S}(x) \in [\underline{C}_x, \overline{C}_x].$$

Notons $\mathcal{V}_E \subseteq \mathcal{V}^*$ l'ensemble des fonctions de valeur potentiellement compatibles avec les exemples E :

$$\mathcal{V}_E = \{v \in \mathcal{V}^* \mid \exists S \in \mathcal{S}^* \mid f_{v,S} \text{ satisfait } E\}.$$

De manière similaire, $\mathcal{S}_E \subseteq \mathcal{S}^*$ est l'ensemble des ensembles de seuils potentiellement compatibles avec les exemples E :

$$\mathcal{S}_E = \{S \in \mathcal{S}^* \mid \exists v \in \mathcal{V}^* \mid f_{v,S} \text{ satisfait } E\}.$$

Enfin, notons $\Omega_E \subseteq \Omega^*$ l'ensemble des paramètres compatibles avec les exemples E :

$$\Omega_E = \{\omega \in \Omega^* \mid f_\omega \text{ satisfait } E\}.$$

Notons que $\Omega_E \subseteq \mathcal{V}_E \times \mathcal{S}_E$.

Définissons $\mathcal{S}_E(v) \subseteq \mathcal{S}_E$ le sous-ensemble de valeurs de seuils possibles compatibles avec les exemples lorsque v est utilisée :

$$\mathcal{S}_E(v) = \{S \in \mathcal{S}^* \mid f_{v,S} \text{ satisfait } E\}.$$

Suivant les principes de la robustesse tels que définis à la section 1.6, nous pouvons maintenant définir des fonctions de tri robustes à partir des fonctions de tri précises, en considérant des ensembles de paramètres au lieu de paramètres précis. Ayant choisi une fonction de valeur $v \in \mathcal{V}^*$, définissons la fonction de tri robuste $f_{v,\mathcal{S}_E(v)} : \forall a \in \mathbb{A}$,

$$f_{v,\mathcal{S}_E(v)}(a) = \bigcup_{S \in \mathcal{S}_E(v)} f_{v,S}(a).$$

Pour rappel, une fonction de tri robuste f_{Ω_E} peut aussi être définie sans fixer de paramètres : $\forall a \in \mathbb{A}$,

$$f_{\Omega_E}(a) = \bigcup_{\omega \in \Omega_E} f_\omega(a).$$

2.3 Méthodes relatives à la problématique de tri

Notons que

$$\bigcup_{v \in \mathcal{V}_E} \bigcup_{S \in \mathcal{S}_E(v)} f_{v,S} = f_{\Omega_E}.$$

La fonction de tri robuste $f_{v, \mathcal{S}_E(v)}$ permet d'obtenir des conclusions en choisissant simplement une fonction de valeur, sans devoir choisir des seuils. Cela réduit le nombre de paramètres à fixer. Les conclusions sont valables quels que soient les seuils. On peut aussi s'intéresser aux conclusions qui peuvent être obtenues sans fixer de fonction de valeur ni de seuils. En suivant encore les principes de la robustesse, on retrouve la fonction de tri robuste déterminée par les exemples : $\forall a \in \mathbb{A}$,

$$\Gamma_P(a) = \bigcup_{v \in \mathcal{V}_E} f_{v, \mathcal{S}_E(v)}(a) = f_{\Omega_E}(a).$$

On peut également chercher un autre type de conclusions :

$$\Gamma_N(a) = \bigcap_{v \in \mathcal{V}_E} f_{v, \mathcal{S}_E(v)}(a).$$

Les fonctions Γ_P et Γ_N indiquent les conclusions possibles et les conclusions nécessaires, considérant les exemples E , concernant les affectations des alternatives. La fonction Γ_N peut également être définie en prenant comme base du raisonnement une fonction de tri imprécise f'_v définie comme suit. Étant donné le choix d'une fonction de valeur v et un ensemble d'exemples E , la fonction f'_v affecte les alternatives à toutes les catégories possibles selon le choix des seuils, de manière compatible avec les exemples E :

$$f'_v(a) = f_{v, \mathcal{S}_E(v)}(a) = \bigcup_{S \in \mathcal{S}_E(v)} f_{v,S}(a).$$

Cette façon de voir permet de manipuler une fonction qui n'a plus qu'un paramètre. Il est alors naturel de définir :

$$\Gamma_N(a) = \bigcap_{v \in \mathcal{V}_E} f'_v(a) = \bigcap_{v \in \mathcal{V}_E} f_{v, \mathcal{S}_E(v)}(a).$$

Ces conclusions nécessaires indiquent les catégories auxquelles les alternatives sont nécessairement affectées, étant donné E mais indépendamment du choix de v . Notons qu'il n'est pas garanti, en toute généralité, que $\Gamma_N(a) \neq \emptyset, \forall a \in \mathbb{A}$. Cette question est étudiée dans l'article cité (Greco *et al.*, 2010).

La méthode proposée par ces auteurs, et sa présentation dans le cadre de ce travail en termes de fonctions robustes et imprécises, génèrent des perspectives intéressantes. Il est clair que les concepts présentés ici n'ont pas grand-chose de

spécifique au modèle à fonction de valeur. Il serait intéressant de généraliser l'exposé et de réfléchir à son application à d'autres classes de modèles. Le lien entre le concept de robustesse et celui de fonction imprécise pourrait être étudié plus avant, et lui aussi généralisé. Il faut remarquer que la fonction de tri imprécise f'_v ne rentre pas parfaitement dans le cadre proposé dans ce travail pour les classes de modèles imprécis (Section 1.5), car la classe de modèle contenant les fonctions f'_v ne peut probablement être définie qu'une fois l'ensemble E fixé. Il serait donc utile de réfléchir à une extension du cadre proposé.

La méthode présentée à la section 3.1 utilise elle aussi l'idée de considérer deux sous-ensembles de paramètres séparément, dans un contexte de décision de groupe.

Les auteurs considèrent aussi la possibilité que le décideur donne des exemples associés à différents intervalles, du plus large au moins large, intervalles associés à différents degrés de crédibilité, à la manière indiquée à la section 2.3.4. Ayant des ensembles d'exemples E_1, E_2, \dots plus ou moins précis, on peut alors obtenir des conclusions plus ou moins fortes à l'aide des fonctions $f_{\Omega_{E_1}}, f_{\Omega_{E_2}}, \dots$. Nous utilisons également le concept d'ensembles d'exemples plus ou moins précis que nous présentons en détail au chapitre 4, raison pour laquelle nous ne détaillons pas cette approche ici. Le chapitre 4 propose de plus une façon simple d'interroger le décideur de façon à obtenir de tels ensembles d'exemples ordonnés par degré de précision.

2.3.3 Désagrégation de modèles de type ÉLECTRE TRI

Mousseau et Słowiński (1998) proposent de trouver les paramètres d'un modèle ÉLECTRE TRI compatibles avec un ensemble d'exemples d'affectations donnés par un décideur. L'approche infère les profils, poids, et seuils, et pas les vétos à cause de la complexité additionnelle. L'analyste choisit si elle infère pour l'affectation pessimiste ou l'optimiste. L'approche utilise un PM non linéaire, à cause de la multiplication des variables de poids avec les valeurs de concordance, valeurs qui elles-mêmes font intervenir les valeurs des profils, inconnus. Nous proposons à la section 3.1 un PLNE pour inférer un modèle proche, de type MR Sort.

Un PL a été proposé de manière à trouver les coalitions gagnantes seulement, les autres paramètres étant supposés connus (Mousseau *et al.*, 2001). De manière similaire, la méthode de Ngo The et Mousseau (2002) trouve les profils, en supposant les autres paramètres déterminés. Dans les deux cas, la connaissance d'une partie du modèle permet de formuler les contraintes de manière linéaire.

Une procédure d'inférence de tous les paramètres d'un modèle ÉLECTRE TRI,

basée sur un algorithme évolutionnaire, a été proposée récemment (Doumpos *et al.*, 2009). L'approche est heuristique, dans le sens où elle ne trouve pas nécessairement un modèle restituant tous les exemples fournis même s'il en existe un, et n'est pas interactive.

Leroy *et al.* (2011) ont testé expérimentalement des outils de désagrégation d'un modèle MR Sort sans véto. Dans une première expérience, ils considèrent un décideur qui fournit des exemples consistants, c'est-à-dire tels qu'il existe un modèle MR Sort qui les satisfait. Les exemples sont générés en utilisant un modèle m_1 de type MR Sort. Ils utilisent un PM pour retrouver un modèle m_2 compatible avec les exemples. Ce n'est bien sûr pas nécessairement le même modèle que celui qui a servi à générer les affectations. L'expérience consiste à observer combien il faut d'exemples d'affectation pour retrouver un modèle m_2 similaire au modèle m_1 de départ. La similitude entre les deux modèles est calculée en déterminant aléatoirement les performances d'un grand nombre d'alternatives et en observant la proportion d'alternatives affectées de la même façon par les deux modèles.

Leur deuxième expérience observe la capacité du PM à trouver un modèle adéquat lorsque des erreurs sont introduites dans les exemples d'affectation. Pour cette expérience, les exemples sont générés par un modèle MR Sort mais sont ensuite bruités : les affectations sont modifiées pour une certaine proportion d'exemples.

Une troisième expérience sert à déterminer dans quelle mesure le PM peut trouver un modèle MR Sort satisfaisant lorsque les affectations des exemples sont déterminées par un modèle basé sur une fonction de valeur additive plutôt que sur un modèle MR Sort.

Le PM utilisé par Leroy *et al.* (2011) est une variante de celui que nous présentons à la section 3.1.3, mais dans un contexte impliquant un seul décideur et sans véto. Par ailleurs, nous proposons à la section 3.2 un algorithme qui permet de calculer plus facilement la similitude entre deux modèles MR Sort sans véto.

2.3.4 Analyse d'inconsistance

Lorsque le décideur fournit un ensemble d'exemples inconsistant, donc qui ne peut pas être représenté dans une classe de modèles de tri M choisie, Mousseau *et al.* (2003) propose (pour certaines classes de modèles) des algorithmes pour calculer l'ensemble d'exemples à retirer pour restaurer la consistance. La méthode a été étendue par la suite pour relâcher les exemples d'affectation plutôt que les effacer (Mousseau *et al.*, 2006). Cette dernière méthode propose au décideur d'associer des degrés de confiance aux exemples, de manière à relâcher de préférence les exemples les moins sûrs. Présentons cette dernière

méthode en détail.

Soit une classe de modèles de tri M associée à un espace de paramètres Ω^* . Supposons que le décideur donne un ensemble d'exemples qu'on souhaite rendre cohérent au regard de la classe M . Il s'agit donc de vérifier s'il existe une fonction de tri $m \in M$ qui satisfasse tous ces exemples d'affectation, et si non, de chercher comment modifier les exemples de manière à restaurer la cohérence. Notons $X \subseteq \mathbb{A}$ l'ensemble des alternatives utilisées en exemples. Un exemple indiquant qu'une alternative $x \in X$ doit être affectée à une catégorie dans l'intervalle $[C_{h_x}, C_{i_x}]$ est noté $(x \rightarrow [C_{h_x}, C_{i_x}])$. Le décideur associe également un degré de confiance à chaque exemple, $v_x \in \mathbb{N}$, indiquant de quels exemples il est le plus enclin à accepter des modifications.

Chaque exemple $(x \rightarrow [C_{h_x}, C_{i_x}])$ définit deux exigences : l'alternative x doit être affectée au moins à C_{h_x} , contrainte notée $(x \succeq C_{h_x})$, et l'alternative x doit être affectée au mieux à C_{i_x} , contrainte notée $(x \preceq C_{i_x})$. Un ensemble d'exemples de ce type définit un ensemble de contraintes sur l'espace des paramètres. Nous ne détaillons pas ces contraintes, qui dépendent de la classe de modèles choisie. Nous en verrons un exemple lors de l'exposé du PM concernant MR Sort. Notons s_X l'ensemble des contraintes contraignant l'espace des paramètres de manière à respecter les exemples concernant les alternatives dans X , et notons Ω_{s_X} l'ensemble des paramètres satisfaisant ces contraintes. Supposons que les exemples définis par le décideur ne puissent être satisfaits par une fonction de tri de M . Cela signifie que Ω_{s_X} est vide. Cherchons des manières de relâcher les contraintes de manière à restaurer la faisabilité. Il est bien sûr souhaitable de relâcher le plus petit nombre de contraintes possibles, et de relâcher de préférence les contraintes associées à un degré de confiance faible.

Supposons que l'on souhaite relâcher uniquement les contraintes de degré de confiance inférieur ou égal à un certain degré de confiance $e \in \mathbb{N}$, en laissant donc les autres contraintes inchangées. Définissons l'ensemble $X_{\leq e}$ contenant les exemples associés à un degré de confiance inférieur ou égal à e , et son complémentaire $X_{>e}$:

$$X_{\leq e} = \{x \in X \mid v_x \leq e\},$$

$$X_{>e} = \{x \in X \mid v_x > e\}.$$

Nous souhaitons donc garder $s_{X_{>e}}$ inchangé, et relâcher certaines contraintes parmi l'ensemble $s_{X_{\leq e}}$.

Une manière de relâcher une contrainte $(x \succeq C_{h_x})$ est d'exiger plutôt $(x \succeq C_{h_x-1})$, ou si nécessaire $(x \succeq C_{h_x-2})$, etc., jusqu'à $(x \succeq C_1)$ qui rend la contrainte triviale. De même, une contrainte $(x \preceq C_{i_x})$ peut être remplacée

par $(x \xrightarrow{\leq} C_{i_x+1})$, voire par $(x \xrightarrow{\leq} C_{i_x+2})$, etc. Définissons donc l'ensemble de contraintes étendues $s_{X_{\leq e}}^{\text{étendues}}$, qui contient toutes les contraintes, originales et relâchées, concernant les exemples de degré de confiance inférieur ou égal à e :

$$s_{X_{\leq e}}^{\text{étendues}} = \bigcup_{x \in X_{\leq e}} \left\{ \begin{array}{l} (x \xrightarrow{\geq} C_{h_x}), (x \xrightarrow{\geq} C_{h_x-1}), (x \xrightarrow{\geq} C_{h_x-2}), \dots, (x \xrightarrow{\geq} C_1), \\ (x \xrightarrow{\leq} C_{i_x}), (x \xrightarrow{\leq} C_{i_x+1}), (x \xrightarrow{\leq} C_{i_x+2}), \dots, (x \xrightarrow{\leq} C_{n_C}) \end{array} \right\}.$$

Cet ensemble de contraintes contient beaucoup de contraintes redondantes. Il s'agit maintenant de chercher dans cet ensemble un nombre minimal de contraintes à effacer pour restaurer la faisabilité.

Pour rechercher un ensemble minimal de contraintes à désactiver parmi $s_{X_{\leq e}}^{\text{étendues}}$, une technique classique en programmation mathématique peut être utilisée : il suffit de définir pour chaque contrainte s de l'ensemble une variable binaire d_s . La variable vaut 1 ssi la contrainte est désactivée. Si la contrainte s s'écrit $\alpha \leq \beta$, elle peut alors être modifiée en la contrainte $\alpha - Md_s \leq \beta$. La valeur M est une constante arbitrairement grande, assurant que la contrainte est nécessairement respectée si d_s prend la valeur 1.

Ayant modifié de la sorte toutes les contraintes de l'ensemble $s_{X_{\leq e}}^{\text{étendues}}$, il ne reste plus qu'à chercher une fonction de tri satisfaisant les contraintes $s_{X_{>e}} \cup s_{X_{\leq e}}^{\text{étendues}}$. La fonction objectif consiste à minimiser le nombre de variables d_s à valeur 1, donc à minimiser la somme des variables d_s .

Une manière d'aider le décideur à résoudre la situation d'inconsistance est de lui présenter un ensemble minimal de modifications d'exemples qui résolvent le problème avec $e = 0$, donc en s'autorisant uniquement à relâcher des exemples associés au degré de confiance le plus faible. Une telle solution n'existe cependant pas nécessairement, ou peut requérir la modification de nombreux exemples. Il est donc également pertinent de calculer des ensembles de modifications minimaux aux niveaux $e = 1, 2, \dots$, de manière à donner au décideur plusieurs possibilités de modifications de ses ensembles d'exemples qui permettent la définition d'un modèle de tri compatible.

Les auteurs ont proposé plusieurs extensions de cette approche. Le décideur peut associer des degrés de confiance différents pour indiquer qu'il préfère qu'un exemple d'affectation soit étendu plutôt vers une catégorie que vers une autre : par exemple il peut préférer transformer $(x \rightarrow [C_{h_x}, C_{i_x}])$ en $(x \rightarrow [C_{h_x-1}, C_{i_x}])$ qu'en $(x \rightarrow [C_{h_x}, C_{i_x+1}])$. Plusieurs jeux de solutions restaurant la cohérence à un niveau de crédibilité e donné peuvent être calculés au lieu d'un seul. Enfin, ils proposent une manière plus générale d'explorer les possibilités de modifications : il est possible d'affecter une pénalité aux exemples relâchés dépendant de leur niveau de confiance et de minimiser la pénalité totale, ce qui peut permettre de trouver par exemple une solution qui contient beaucoup de modifications à un niveau très faible et une seule modification à un niveau

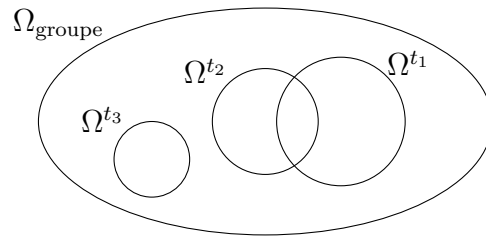


FIGURE 2.5 – Une configuration possible pour les espaces de paramètres Ω^t compatibles avec les exemples du décideur t , $t \in \{t_1, t_2, t_3\}$, ainsi que l'espace de paramètre Ω_{groupe} défini par les exemples du groupe.

de confiance élevé.

2.3.5 Désagrégation pour des groupes

Damart *et al.* (2007) propose une procédure d'aide à la décision pour un groupe de décideurs, avec pour objectif d'aboutir à un modèle de groupe aussi précis que possible. L'article est présenté dans l'hypothèse que la classe de modèles choisie est ÉLECTRE TRI, mais comme le font remarquer les auteurs, elle peut s'appliquer avec n'importe quelle classe de modèles. Il est utile de présenter cette procédure en détail, parce que les outils proposés dans cette thèse en font usage. Nous présentons ici la procédure de façon générale, en notant M la classe de modèles choisie, associée à un espace de paramètres Ω^* .

Lors de la phase d'initialisation, chaque décideur fournit un ensemble d'exemples cohérent pour M . Les outils de résolution d'inconsistance vus ci-dessus peuvent être utilisés par chaque décideur individuellement pour y parvenir. Pour chaque décideur $t \in \mathcal{T}$, notons Ω^t l'ensemble de modèles satisfaisant les exemples de t . Ensuite, la procédure définit itérativement un modèle de groupe Ω_{groupe} de plus en plus précis en y ajoutant des exemples progressivement. La procédure vise à maintenir la cohérence des modèles individuels et du modèle de groupe à chaque itération : chaque fois qu'un exemple est ajouté au modèle de groupe, l'exemple est également ajouté à chaque modèle individuel. Ainsi, le processus garantit que le modèle de chaque individu est au moins aussi précis que le modèle de groupe, au début de chaque itération. La figure 2.5 illustre ce fait. Une itération fonctionne de la façon suivante.

1. L'analyste, sur base des exemples individuels et de groupe définis au stade actuel, propose un ou plusieurs exemples d'affectation, donc une paire alternative et sous-ensemble de catégories, candidats à être ajoutés au modèle de groupe.

2. Les décideurs peuvent vérifier individuellement, à l'aide des outils proposés ci-dessus, si l'ajout d'un candidat exemple est compatible avec leurs exemples existants. Si l'exemple contredit leur modèle individuel à ce stade, ils peuvent examiner les façons possibles de rétablir la cohérence, et en fonction de cet examen décider de leur accord à l'ajout de cet exemple au modèle de groupe.
3. Lorsque un exemple sur lequel les décideurs s'accordent a été trouvé, il est ajouté au modèle de groupe et aux modèles de chaque décideur individuel. Les décideurs individuels résolvent l'incohérence éventuelle de leur modèle propre en modifiant leurs exemples individuels avant de poursuivre. Ils doivent cependant conserver les exemples du groupe afin de maintenir la compatibilité de leur modèle individuel avec le modèle de groupe.

Au fur et à mesure des itérations, le modèle de groupe se précise. Le processus s'arrête soit lorsque les décideurs considèrent que le modèle de groupe obtenu est satisfaisant, soit parce qu'ils ne sont plus capables de s'accorder sur un exemple à ajouter. Le groupe peut également décider à une certaine étape de remettre en cause un exemple ajouté précédemment au modèle de groupe.

L'article propose également deux indicateurs pour guider l'analyste lors du choix des candidats exemples à la première étape de l'itération. Les indicateurs visent à trouver les exemples les plus susceptibles de susciter un accord du groupe et considèrent la proportion de décideurs qui ont déjà cet exemple dans leurs exemples individuels, et la difficulté que cela représenterait pour chaque décideur d'ajouter cet exemple, fonction de l'ampleur des changements à effectuer dans son modèle individuel pour restaurer la cohérence après ajout de l'exemple.

L'article postule, dans le cas où la classe de modèles choisie est ÉLECTRE TRI, que les seuils de veto, de préférence, d'indifférence et les profils sont fixés au préalable, laissant donc uniquement les coalitions libres. Deux raisons peuvent justifier ce choix. D'une part, au moment où l'article est paru, les techniques de désagrégation ne proposaient pas de manière performante de désagréger l'entièreté du modèle, comme indiqué ci-dessus. D'autre part, la procédure fonctionnera vraisemblablement mieux si l'ensemble de départ des paramètres possibles n'est pas trop vaste, ceci afin d'éviter de devoir demander aux décideurs de nombreux exemples afin de contraindre le modèle.

La table 2.1 résume les fonctionnalités principales des procédures d'élicitation proposées dans le cadre de la méthode ÉLECTRE TRI. Pour chaque article, la deuxième colonne indique l'entrée attendue de l'outil d'élicitation proposé. La dernière colonne montre le résultat de l'outil.

Nous proposons à la section 3.1 une procédure permettant d'aboutir à un

Article	Entrée	Sortie
Mousseau et Słowiński (1998)	i	L, W (programme non linéaire)
Mousseau <i>et al.</i> (2001)	i, L	W
Ngo The et Mousseau (2002)	i, W	L
Dias <i>et al.</i> (2002)	i	modèle robuste (L, W)
Mousseau <i>et al.</i> (2003)	i^*	exemples à retirer
Mousseau <i>et al.</i> (2006)	i^*	exemples à relâcher
Damart <i>et al.</i> (2007)	g, L	modèle de groupe (W)

TABLE 2.1 – Procédures d’inférence pour modèles de type ÉLECTRE TRI. Le symbole i désigne des exemples d’affectation d’un seul décideur, i^* désigne des exemples d’affectation d’un seul décideur potentiellement incohérents, g est un ensemble d’ensembles d’exemples d’affectation associés à un groupe de décideurs, L est un ensemble de profils, W désigne un ensemble de coalitions.

ensemble de profils partagés par le groupe, sans déterminer de valeurs pour les poids. En l’associant à Damart *et al.* (2007), cette procédure permet d’obtenir un modèle de groupe sans supposer la connaissance de paramètres initiaux d’une partie du modèle ÉLECTRE TRI.

Deuxième partie

Contributions méthodologiques

Tous les ministres avaient un profond respect pour les oracles; tous convenaient ou feignaient de convenir qu'ils étaient le fondement de la religion; que la raison doit se taire devant eux; que c'est par eux que les rois règnent sur les peuples, et les mages sur les rois; que sans les oracles il n'y aurait ni vertu ni repos sur la terre. Enfin, après avoir témoigné la plus profonde vénération pour eux, presque tous conclurent que celui-ci était impertinent, qu'il ne fallait pas lui obéir; (...)

(La princesse de Babylone, Voltaire (2000, p. 19))

3 Convergence de modèles MR

Sort pour la décision de groupe

Ce chapitre présente des outils s'adressant à un groupe de décideurs qui souhaitent se mettre d'accord sur une procédure d'évaluation. La section 3.1 présente une méthode visant à construire progressivement un modèle de groupe, basé sur MR Sort, à partir d'exemples individuels. La méthode est illustrée à l'aide d'un exemple fictif, et les performances des algorithmes proposés sont testés. La section 3.2 propose des algorithmes pour calculer la différence entre deux modèles MR Sort et discute de l'utilité de cette mesure lors du travail avec un groupe. La section 3.3 présente une application réelle de certains des outils proposés, application qui visait à construire un modèle de tri de groupe pour l'évaluation de dangerosité de substances polluantes.

3.1 Élicitation de paramètres pour méthode MR Sort

Cette section propose des outils visant à construire progressivement un modèle de tri pour un groupe de décideurs en utilisant une approche basée sur la désagrégation. Le contenu de cette section est repris d'un article paru dans *European Journal of Operational Research (EJOR)* (Cailloux *et al.*, 2012b), à la différence que les algorithmes d'inférence sont ici décrits dans le cas plus général où les décideurs peuvent donner des exemples d'affectation à des intervalles de catégories au lieu d'exemples à des catégories précises.

3.1.1 Contexte et objectif

Supposons le contexte suivant. Les décideurs d'un ensemble \mathcal{T} de décideurs souhaitent s'accorder sur un moyen d'évaluer n'importe quelle alternative d'un ensemble d'alternatives possibles \mathbb{A} , c'est-à-dire d'associer à chaque alternative $a \in \mathbb{A}$ une catégorie parmi un ensemble \mathcal{C} donné a priori de catégories ordonnées par préférence. Les décideurs s'accordent sur un ensemble de critères \mathcal{J} à utiliser pour évaluer les alternatives et s'accordent sur les performances a_j

3 Convergence de modèles MR Sort pour la décision de groupe

de chaque alternative sur chaque critère. Chaque décideur $t \in \mathcal{T}$ fournit des exemples d'affectation qui serviront de base à la modélisation initiale de ses préférences. L'ensemble des exemples d'affectation du décideur t est noté E^t . Un exemple d'affectation est une association d'une alternative $a \in \mathbb{A}$ et d'un intervalle de catégories $I \in \mathcal{C}_I$, et est noté $(a \xrightarrow{t} I)$. Cela représente le fait que le décideur t trouve que l'alternative a devrait être affectée à une catégorie C parmi I , et ne devrait pas être affectée à une catégorie hors de cette intervalle. L'ensemble des alternatives utilisées comme exemples est noté X . On a donc $X \subseteq \mathbb{A}$.

L'objectif des décideurs est de définir une fonction de tri consensuelle, c'est-à-dire une fonction de tri que tous les décideurs considèrent acceptable. Notons d'emblée que dans certaines situations, cet objectif n'est pas accessible : les raisonnements des décideurs pourraient être trop différents pour s'accorder. Nous supposons cependant que les décideurs sont prêts à écouter le point de vue des autres décideurs et peuvent envisager de changer d'avis si nécessaire pour parvenir à un consensus. Les outils que nous proposons s'appuient sur cette hypothèse et fournissent des manières de faciliter la discussion pour aider le groupe de décideurs à parvenir à un consensus. S'il s'avère que les décideurs raisonnent de façons trop différentes pour pouvoir parvenir à un modèle consensuel, l'approche que nous proposons peut être utile néanmoins car elle peut apporter des informations concernant la nature de ces divergences. Par ailleurs, nous nous situons dans un contexte de collaboration, et non de négociation, nous ne nous soucions donc pas de questions de manipulabilité de la procédure.

Ce chapitre introduit de nouveaux outils d'inférence, et indique comment ces outils peuvent se combiner aux outils existants dans un processus aboutissant à un consensus parmi les décideurs, dans le cas où ils sont prêts à changer suffisamment d'avis pour rendre cette issue possible. Nous nous basons sur la classe de modèles MR Sort et ne postulons pas qu'une partie du modèle de préférence a déjà été établie.

L'approche proposée se base, de manière classique, sur la recherche d'un modèle de tri reproduisant les exemples d'affectation donnés. Un modèle de tri ω , déterminant une fonction de tri f_ω , satisfait un ensemble d'exemples d'affectation E^t d'un décideur $t \in \mathcal{T}$ ssi

$$\forall (a \xrightarrow{t} I) \in E^t : f_\omega(a) \in I.$$

Notre approche présente deux différences avec les approches de désagrégation existantes mentionnées à la section 2.3.3.

D'une part, nous formulons la recherche d'un tel modèle comme un PLNE. Les méthodes de désagrégation existantes pour les modèles de type ÉLECTRE TRI

3.1 Élicitation de paramètres pour méthode MR Sort

étaient soit moins performantes (Mousseau et Słowiński, 1998), soit postulaient qu’une partie des paramètres du modèle de préférence, les poids ou les profils, était déjà spécifiée. Une exception est l’article de Leroy *et al.* (2011), qui utilise la même approche que celle présentée ici pour représenter les contraintes de façon linéaire, mais dans un cadre mono-décideur.

D’autre part, nous scindons le problème de trouver un modèle consensuel en deux étapes, en proposant de chercher dans un premier temps un ensemble de profils partagé tels qu’il est possible de trouver pour chaque décideur un ensemble de coalitions gagnantes qui restitue ses exemples. Nous désignons sous le nom d’ensemble de coalitions de critères gagnantes, ou simplement coalitions gagnantes, les valeurs des paramètres constitués des poids de chaque critère et du seuil de coupe. Cette étape fournit un modèle partiellement consensuel, où les valeurs des profils sont partagées mais pas les valeurs des poids ni des seuils de coupe. La procédure de Damart *et al.* (2007), détaillée à la section 2.3.5, peut ensuite être utilisée pour parvenir à un modèle de préférence consensuel. L’approche proposée s’intègre donc avec les outils existants pour fournir une méthode complète de résolution du problème posé, et suggère une façon de diviser un problème complexe en deux sous-problèmes plus simples. En effet, s’assurer de l’accord du groupe de décideurs concernant une partie du modèle avant de passer à la suite de l’élicitation peut augmenter la stabilité du processus dans son ensemble.

Nous proposerons les outils de calculs suivants, sous forme de PMs.

IMRS *Infer MR Sort* infère, si possible, un modèle MR Sort complet (valeurs des profils, des poids et des seuils de coupe), satisfaisant un ensemble d’exemples d’affectations donné.

ICL *Infer Category Limits* infère, si possible, les limites des catégories, ou profils, d’un modèle MR Sort, de telle sorte que, pour chaque décideur t , il existe des coalitions telles que le modèle MR Sort résultant des profils partagés et de ces coalitions satisfasse les exemples donnés par le décideur.

ICLV *Infer Category Limits with Vetoes* étend le programme précédent en ajoutant l’inférence de seuils de vétos. Il infère, si possible, des valeurs pour les profils et seuils de vétos partagés, et des coalitions éventuellement distinctes pour chaque décideur, satisfaisant les exemples de chaque décideur.

CWR *Compute Weights Restriction* donne une indication sur la contrainte que le choix d’un jeu de valeurs pour les profils impose à un décideur donné quand au choix des poids.

Les trois premiers programmes sont de proches variantes d’un même PM, mais ils résolvent des problèmes différents. Il est donc commode de leur donner

des dénominations distinctes auxquelles nous nous référons dans la suite du travail. Nous ne présentons pas dans la suite le programme IMRS en détail, car il s'obtient facilement à partir de ICL en n'utilisant qu'un jeu de variables pour les coalitions au lieu d'en créer une version pour chaque décideur. Nous présentons ICL avant ICLV, bien que ICLV puisse être considéré comme une généralisation de ICL, afin de réduire la complexité de chaque partie.

Une bibliothèque implémentant ces différents programmes mathématiques, J-MCDA, est publiée en tant que logiciel libre. La bibliothèque permet de facilement obtenir les quatre programmes présentés ici ainsi que d'autres variantes. Par exemple, il est possible de résoudre un programme IMRS qui tient également compte de la présence de vétos. J-MCDA est présentée à la section 6.2.

3.1.2 Un exemple de processus d'aide à la décision

Les outils développés ne constituent qu'une partie des moyens à disposition de l'analyste qui souhaite aider un groupe de décideurs. Cette section présente un exemple de processus d'aide à la décision afin d'illustrer la combinaison possible des outils proposés et des outils existants présentés à la section 2.3.

Les étapes suivantes peuvent être suivies en vue d'obtenir des profils consensuels.

- Demander à chaque décideur des exemples d'affectation personnels.
- Chercher un modèle consensuel complet avec IMRS, en utilisant l'union des exemples d'affectation de tous les décideurs.
- S'il n'existe pas de tel modèle consensuel, chercher à l'aide de ICL un jeu de valeurs de profils partagé qui satisfait les exemples lorsque des coalitions individuelles sont utilisées.
- S'il n'est pas possible de trouver des profils partagés, ICLV peut être utilisé pour trouver des profils et des vétos partagés.
- S'il est souhaité de trouver un modèle sans vétos, ou si l'autorisation de vétos ne permet pas de trouver de profils partagés, les programmes proposés par Mousseau *et al.* (2003, 2006) (cf. Table 2.1 page 66) peuvent être utilisés pour trouver les exemples conduisant à des inconsistences, soit au niveau de chaque décideur individuellement, soit en considérant le groupe comme un individu, donc en considérant l'union des exemples d'affectation. L'analyste peut alors proposer aux décideurs concernés de modifier leurs exemples.
- Lorsqu'une solution est trouvée, calculer grâce à CWR, pour chaque décideur, la réduction de latitude que ces valeurs de profils entraînent concernant le choix des poids.

3.1 Élicitation de paramètres pour méthode MR Sort

- Demander aux décideurs s'ils acceptent le choix de ces profils et les conséquences sur le choix des poids. Si non, inclure des contraintes supplémentaires et relancer les programmes ICL ou ICLV.
- À n'importe quel moment durant le processus, les décideurs peuvent ajouter des contraintes au programme d'inférence, par exemple en imposant une borne sur une valeur de profil ou en imposant une valeur pour un seuil de véto.

Nous proposons de commencer par chercher un modèle complet avec IMRS. En effet, cela revient à chercher non seulement des profils partagés mais également des coalitions partagées. Si un tel modèle satisfaisant tous les exemples existe, il n'y a pas d'inconvénient à utiliser celui-là plutôt qu'une solution du programme ICL. L'analyste peut alors choisir de présenter le modèle résultant aux décideurs dans l'espoir de valider tous les paramètres à la fois. Si le groupe accepte ce modèle, le problème est résolu et il n'est pas nécessaire d'utiliser de méthodes plus subtiles. Cela n'empêche pas l'analyste, si elle le désire (ou dans le cas où les décideurs ne valident pas le modèle complet trouvé), d'avancer plus prudemment en se concentrant dans un premier temps sur les valeurs des profils. Nous pensons qu'une telle approche réduira en général le délai de convergence du processus, en fournissant une étape intermédiaire acceptée comme base par les décideurs (le choix des profils) et qui servira alors de point d'appui pour la deuxième phase du processus visant à l'obtention de coalitions partagées. De plus, l'existence d'un modèle consensuel complet satisfaisant tous les exemples, cherché par IMRS, est nettement moins probable que l'existence d'une solution à ICL. Nous postulons donc que l'approche proposée ici s'avèrera utile dans de nombreux cas pratiques.

Une fois que les valeurs communes des profils sont obtenues, l'approche proposée par Damart *et al.* (2007) permet de faire converger le processus vers l'obtention de coalitions partagées, en proposant itérativement aux décideurs de modifier des exemples bien choisis.

3.1.3 ICL : Infer Category Limits

Soit un ensemble d'alternatives $X \subseteq \mathbb{A}$ utilisées comme exemples d'affectation, un ensemble d'indices de critères \mathcal{J} , les évaluations des alternatives exemples $a_j, \forall a \in X, j \in \mathcal{J}$, un ensemble de catégories $\mathcal{C} = \{C_1, \dots, C_n\}$, un ensemble de profils associés $L = \{l^-(C), \forall C \in \mathcal{C} \setminus C_1\}$, un ensemble de décideurs \mathcal{T} , des exemples d'affectation $E^t, \forall t \in \mathcal{T}$. Le programme ICL détermine les performances des profils $l \in L$, partagés entre les décideurs, et des valeurs individuelles de poids $w_j^t, \forall j \in \mathcal{J}$ et de seuils de majorité λ^t , pour chaque décideur $t \in \mathcal{T}$, satisfaisant tous les exemples d'affectation. Rappelons qu'un profil est noté simplement l lorsque les catégories que le profil sépare ne sont

3 Convergence de modèles MR Sort pour la décision de groupe

pas importantes, ou $l^-(C)$ ou $l^+(C)$ pour représenter la limite inférieure ou supérieure de la catégorie C .

Le PM définit donc les trois types de variables de décision : performance des profils, poids et seuils de majorité. Il requiert aussi des variables techniques, nécessaires pour s'assurer que le modèle trouvé satisfait les exemples d'affectation. Voici les variables utilisées dans le PM ICL.

Performance des profils $l_j, \forall l \in L, j \in \mathcal{J}$: la performance, sur le critère j , du profil l . Cette variable est continue.

Poids $w_j^t, \forall t \in \mathcal{T}, j \in \mathcal{J}$: le poids du critère j , pour le décideur t . Il s'agit d'une variable continue non négative.

Seuils de majorité $\lambda^t, \forall t \in \mathcal{T}$: le seuil de majorité pour le décideur t , variable continue dans $[1/2, 1]$.

Concordance partielle binaire $K_j(a, l^-(C)), \forall a \in X, C \in \mathcal{C} \setminus C_1, j \in \mathcal{J}$: la concordance partielle, du point de vue du critère j , indiquant si l'alternative a mérite d'atteindre la catégorie C , ou de manière équivalente, indiquant si l'alternative a est au moins aussi bonne que le profil $l^-(C)$. La variable $K_j(a, l^-(C))$ vaut 1 si la performance de l'alternative a sur le critère j est au moins aussi bonne que la performance du profil $l^-(C)$, 0 sinon. Elle est définie comme une variable binaire du PM.

$$K_j(a, l^-(C)) = \begin{cases} 1 & \Leftrightarrow a_j \geq l_j^-(C) \\ 0 & \text{sinon.} \end{cases}$$

Concordance partielle pondérée $\Psi_j^t(a, l^-(C)), \forall t \in \mathcal{T}, a \in X, C \in \mathcal{C} \setminus C_1, j \in \mathcal{J}$: une variable équivalente à la variable de concordance partielle binaire, mais pondérée par le poids w_j^t . Sa valeur est $\Psi_j^t(a, l^-(C)) = w_j^t K_j(a, l^-(C))$. Cette variable vaut donc w_j^t si la performance de l'alternative a sur le critère j est au moins aussi bonne que la performance du profil $l^-(C)$, 0 sinon. C'est une variable continue non négative.

$$\Psi_j^t(a, l^-(C)) = \begin{cases} w_j^t & \Leftrightarrow a_j \geq l_j^-(C) \\ 0 & \text{sinon.} \end{cases}$$

Slack $s \geq 0$: une variable slack, utilisée uniquement dans une variante du programme.

Contraintes

Une première série de contraintes permet de s'assurer que la somme des poids vaut 1 : $\forall t \in \mathcal{T}$,

$$\sum_{j \in \mathcal{J}} w_j^t = 1. \quad (\text{somme poids})$$

3.1 Élicitation de paramètres pour méthode MR Sort

Ces contraintes imposent également $w_j^t \leq 1, \forall t \in \mathcal{T}, j \in \mathcal{J}$, car ces variables sont non négatives par définition. Cela sera utile ultérieurement.

Les contraintes suivantes assurent un ordonnancement correct des profils définissant les catégories, $\forall C \in \mathcal{C} \setminus \{C_1, C_n\}, j \in \mathcal{J}$:

$$l_j^-(C) \leq l_j^+(C). \quad (\text{profils croissants})$$

Le PM doit être défini de manière à s'assurer que le modèle de préférence trouvé satisfasse les exemples. Il faut donc s'assurer des deux inégalités suivantes, $\forall t \in \mathcal{T}, \forall (a \xrightarrow{t} [\underline{C}, \overline{C}]) \in E^t$:

$$\sum_{j \in \mathcal{J}: a_j \geq l_j^-(\underline{C})} w_j^t \geq \lambda^t \text{ si } \underline{C} \neq C_1, \quad (3.1)$$

et

$$\sum_{j \in \mathcal{J}: a_j \geq l_j^+(\overline{C})} w_j^t < \lambda^t \text{ si } \overline{C} \neq C_n. \quad (3.2)$$

L'inégalité (3.1) garantit l'affectation de l'alternative à une catégorie au moins aussi bonne que \underline{C} , et l'inégalité (3.2) empêche une affectation à une catégorie meilleure que \overline{C} . Ces contraintes peuvent être formulées à l'aide des variables $\Psi_j^t(a, l)$, car $\sum_{j \in \mathcal{J}} \Psi_j^t(a, l) = \sum_{j \in \mathcal{J}: a_j \geq l_j} w_j^t$. Les contraintes s'expriment alors comme suit, $\forall t \in \mathcal{T}, \forall (a \xrightarrow{t} [\underline{C}, \overline{C}]) \in E^t$:

$$\sum_{j \in \mathcal{J}} \Psi_j^t(a, l^-(\underline{C})) \geq \lambda^t \text{ si } \underline{C} \neq C_1, \quad (\text{support plancher})$$

et

$$\sum_{j \in \mathcal{J}} \Psi_j^t(a, l^+(\overline{C})) < \lambda^t \text{ si } \overline{C} \neq C_n. \quad (\text{support plafond})$$

Il reste à définir les variables $\Psi_j^t(a, l)$. Commençons par définir $K_j(a, l)$, qui indique si $a_j \geq l_j$. $\forall j \in \mathcal{J}, a \in X, l \in L$:

$$\frac{1}{M}(a_j - l_j) < K_j(a, l) \leq \frac{1}{M}(a_j - l_j) + 1. \quad (\text{concordance binaire})$$

La constante M est une valeur arbitrairement grande garantissant $-1 < \frac{1}{M}(a_j - l_j) < 1$. Étant donné que les variables $K_j(a, l)$ sont définies binaires, ces contraintes garantissent que

$$a_j \geq l_j \Rightarrow K_j(a, l) = 1,$$

3 Convergence de modèles MR Sort pour la décision de groupe

et que

$$a_j < l_j \Rightarrow K_j(a, l) = 0.$$

En effet, $[a_j \geq l_j] \Rightarrow [a_j - l_j \geq 0] \Rightarrow [\frac{1}{M}(a_j - l_j) + 1 \geq 1]$, donc $K_j(a, l) \leq \frac{1}{M}(a_j - l_j) + 1$ est nécessairement satisfait indépendamment de la valeur de $K_j(a, l)$, et $[a_j \geq l_j] \Rightarrow [0 \leq \frac{1}{M}(a_j - l_j) < 1]$, donc $K_j(a, l)$ est contraint à évaluer 1.

De manière similaire, $[a_j - l_j < 0] \Rightarrow [-1 < \frac{1}{M}(a_j - l_j) < 0]$, donc $\frac{1}{M}(a_j - l_j) < K_j(a, l)$ est nécessairement satisfait, et $0 < \frac{1}{M}(a_j - l_j) + 1 < 1$ donc $K_j(a, l)$ est contraint à une valeur de 0.

Une fois que $K_j(a, l)$ est définie, le plus simple pour définir $\Psi_j^t(a, l)$ serait d'utiliser l'expression $\Psi_j^t(a, l) = w_j^t K_j(a, l)$, mais la contrainte ainsi définie ne serait pas linéaire étant donné que ces deux valeurs sont inconnues. Les contraintes suivantes remplacent cette expression, suivant une idée proposée par Meyer *et al.* (2008), $\forall t \in \mathcal{T}, j \in \mathcal{J}, a \in X, l \in L$:

$$\begin{aligned} w_j^t + K_j(a, l) - 1 &\leq \Psi_j^t(a, l) \leq K_j(a, l) && \text{(concordance pondérée 1)} \\ \Psi_j^t(a, l) &\leq w_j^t. && \text{(concordance pondérée 2)} \end{aligned}$$

Rappelons que $\Psi_j^t(a, l) \geq 0$ et $w_j^t \leq 1$, donc $w_j^t - 1 \leq 0$. Dès lors, ces contraintes garantissent que $\Psi_j^t(a, l) = 0 \Leftrightarrow K_j(a, l) = 0$ et $\Psi_j^t(a, l) = w_j^t \Leftrightarrow K_j(a, l) = 1$. En effet, $[K_j(a, l) = 0] \Rightarrow [w_j^t + K_j(a, l) - 1 \leq 0] \Rightarrow \Psi_j^t(a, l) = 0$ (par (concordance pondérée 1)), et $[K_j(a, l) = 1] \Rightarrow [w_j^t \leq \Psi_j^t(a, l) \leq 1]$ (par (concordance pondérée 1)), donc $[w_j^t \leq \Psi_j^t(a, l) \leq w_j^t]$ (grâce à (concordance pondérée 2)).

Le PM à résoudre est résumé dans le PM 1, où les inégalités strictes ont été remplacées par des inégalités larges à l'aide d'une valeur positive arbitrairement petite ε .

Séparation entre les poids et les seuils de majorité

Le programme présenté n'utilise pas de fonction objectif : n'importe quelle solution de l'espace défini par les contraintes convient. Un objectif peut être utilisé pour, par exemple, maximiser l'écart entre les sommes de poids et les seuils de majorité. Une simple modification des contraintes (support plancher) et (support plafond) suffit, en introduisant une nouvelle variable slack s continue non négative. La fonction objectif consiste alors à maximiser s .

$$\forall t \in \mathcal{T}, \forall (a \xrightarrow{t} [\underline{C}, \overline{C}]) \in E^t :$$

$$\sum_{j \in \mathcal{J}} \Psi_j^t(a, l^-(\underline{C})) \geq \lambda^t + s \text{ si } \underline{C} \neq C_1,$$

3.1 Élicitation de paramètres pour méthode MR Sort

Trouver une solution satisfaisant

$$\begin{array}{l}
 \forall t \in \mathcal{T} : \\
 \forall C \in \mathcal{C} \setminus \{C_1, C_n\}, j \in \mathcal{J} : \\
 \forall a \in X, l \in L, j \in \mathcal{J} : \\
 \forall t \in \mathcal{T}, a \in X, l \in L, j \in \mathcal{J} : \\
 \forall t \in \mathcal{T}, \forall (a \xrightarrow{t} [\underline{C}, \overline{C}]) \in E^t, \underline{C} \neq C_1 : \\
 \forall t \in \mathcal{T}, \forall (a \xrightarrow{t} [\underline{C}, \overline{C}]) \in E^t, \overline{C} \neq C_n :
 \end{array}
 \left. \begin{array}{l}
 \sum_{j \in \mathcal{J}} w_j^t = 1. \\
 l_j^-(C) \leq l_j^+(C). \\
 \frac{(a_j - l_j) + \varepsilon}{M} \leq K_j(a, l) \leq \frac{a_j - l_j}{M} + 1. \\
 \left\{ \begin{array}{l} w_j^t + K_j(a, l) - 1 \leq \Psi_j^t(a, l) \leq K_j(a, l) \\ \Psi_j^t(a, l) \leq w_j^t. \end{array} \right. \\
 \sum_{j \in \mathcal{J}} \Psi_j^t(a, l^-(\underline{C})) \geq \lambda^t. \\
 \sum_{j \in \mathcal{J}} \Psi_j^t(a, l^+(\overline{C})) \leq \lambda^t - \varepsilon.
 \end{array} \right\} (E)$$

PM 1: ICL

et

$$\sum_{j \in \mathcal{J}} \Psi_j^t(a, l^+(\overline{C})) + s < \lambda^t \text{ si } \overline{C} \neq C_n.$$

3.1.4 ICLV : Inferring Category Limits with Vetoes

Supposons que l'utilisation de vétos soit admise, donc que le PM puisse utiliser des seuils de vétos lors de la recherche de valeurs de profils partagés visant à satisfaire les exemples. Cette situation est plus générale que la précédente, en le sens que s'il existe une solution pour ICL, il existe une solution pour ICLV. En effet, l'usage de vétos n'est pas obligatoire : ce nouveau PM a simplement l'opportunité supplémentaire par rapport au programme précédent d'utiliser des seuils de vétos. Le PM présenté ici suppose que les décideurs partagent les valeurs des vétos. Il est facile de transformer ce programme pour lui faire trouver des vétos individuels, éventuellement distincts, pour chaque décideur. Mentionnons toutefois qu'un tel modèle présenterait un risque de surajustement (*overfitting*) important et pourrait s'avérer trop général pour être utile.

Nous ne détaillons que les modifications par rapport au PM ICL. En particulier, l'ensemble de contraintes marquées (E) dans le PM 1 page 77 est commun aux deux programmes.

Les valeurs des seuils de vétos sont représentées par des variables continues $\nu_j^{\text{BM}}(C)$, associées à chaque critère $j \in \mathcal{J}$ et catégorie $C \in \mathcal{C} \setminus \{C_1\}$. Ces variables représentent les seuils de véto au sens indiqué en section 2.2.3, soit $\nu_j^{\text{BM}}(C) = l_j^-(C) - v_j$, donc une valeur en-dessous de laquelle l'alternative est interdite d'accès en catégorie C ou mieux. Les variables binaires $V_j(a, C), \forall j \in$

3 Convergence de modèles MR Sort pour la décision de groupe

$\mathcal{J}, a \in X, C \in \mathcal{C} \setminus \{C_1\}$, indiquent si le critère j impose un veto à l'accès à C de a :

$$V_j(a, C) = D_j^{\text{binaire}}(a, l^-(C)) = \begin{cases} 1 & \Leftrightarrow a_j < \nu_j^{\text{BM}}(C), \\ 0 & \text{sinon.} \end{cases}$$

Contraintes supplémentaires

Il faut s'assurer que les vetos sont inférieurs au profil correspondant, $\forall C \in \mathcal{C} \setminus \{C_1\}, j \in \mathcal{J}$:

$$\nu_j^{\text{BM}}(C) \leq l_j^-(C). \quad (\text{veto, plafond})$$

L'ordre sur les seuils de veto est respecté grâce aux contraintes suivantes, $\forall C_h \in \mathcal{C} \setminus \{C_1, C_n\}, j \in \mathcal{J}$:

$$\nu_j^{\text{BM}}(C_h) \leq \nu_j^{\text{BM}}(C_{h+1}). \quad (\text{veto, ordre})$$

Les variables binaires $V_j(a, C)$ sont définies comme suit, $\forall a \in X, C \in \mathcal{C} \setminus \{C_1\}, j \in \mathcal{J}$:

$$\frac{\nu_j^{\text{BM}}(C) - a_j}{M} \leq V_j(a, C) < \frac{\nu_j^{\text{BM}}(C) - a_j}{M} + 1. \quad (\text{veto binaire})$$

Ceci garantit que $V_j(a, C) = 1 \Leftrightarrow a_j < \nu_j^{\text{BM}}(C)$, par un raisonnement similaire à celui tenu pour les contraintes (concordance binaire).

Les contraintes (support plancher), (support plafond) doivent être redéfinies pour tenir compte des vetos, $\forall t \in \mathcal{T}, \forall (a \xrightarrow{t} [\underline{C}_h, \overline{C}_i]) \in E^t$:

$$\sum_{j \in \mathcal{J}} \Psi_j^t(a, l^-(\underline{C}_h)) \geq \lambda^t + \sum_{j \in \mathcal{J}} V_j(a, \underline{C}_h) \text{ si } \underline{C}_h \neq C_1, \quad (\text{support plancher avec veto})$$

et

$$\sum_{j \in \mathcal{J}} \Psi_j^t(a, l^+(\overline{C}_i)) < \lambda^t + \sum_{j \in \mathcal{J}} V_j(a, \overline{C}_{i+1}) \text{ si } \overline{C}_i \neq C_n. \quad (\text{support plafond avec veto})$$

Notons que $\sum_{j \in \mathcal{J}} V_j(a, C) \geq 1$ ssi au moins un critère place son veto et barre l'accès à C pour l'alternative a . Ces contraintes servent, comme précédemment, à assurer la satisfaction des exemples d'affectation. Lorsque le décideur indique qu'une alternative a doit aller au moins en catégorie \underline{C}_h , la contrainte (support plancher avec veto) assure qu'aucun veto n'empêche l'accès de a à \underline{C}_h . En effet, il est impossible de satisfaire la contrainte s'il existe un critère j tel que $V_j(a, l^-(\underline{C}_h)) = 1$. Lorsque le décideur indique que a ne peut être affectée à une catégorie meilleure que \overline{C}_i , la contrainte (support plafond avec veto) requiert, soit (comme précédemment) que le support en faveur de l'alternative

3.1 Élicitation de paramètres pour méthode MR Sort

$\min \sum_{j \in \mathcal{J}} V_j$ t.q.

$$\left\{ \begin{array}{l} \text{Contraintes (E) indiquées au PM 1} \\ \forall t \in \mathcal{T}, \forall (a \xrightarrow{t} [\underline{C}, \overline{C}]) \in E^t, \underline{C} \neq C_1 : \quad \sum_{j \in \mathcal{J}} \Psi_j^t(a, l^-(\underline{C})) \geq \lambda^t + \sum_{j \in \mathcal{J}} V_j(a, \underline{C}). \\ \forall t \in \mathcal{T}, \forall (a \xrightarrow{t} [\underline{C}, \overline{C}_i]) \in E^t, \overline{C}_i \neq C_n : \quad \sum_{j \in \mathcal{J}} \Psi_j^t(a, l^+(\overline{C}_i)) \leq \lambda^t + \sum_{j \in \mathcal{J}} V_j(a, \overline{C}_{i+1}) - \varepsilon. \\ \forall C \in \mathcal{C} \setminus \{C_1\}, j \in \mathcal{J} : \quad \nu_j^{\text{BM}}(C) \leq l_j^-(C). \\ \forall C_h \in \mathcal{C} \setminus \{C_1, C_n\}, j \in \mathcal{J} : \quad \nu_j^{\text{BM}}(C_h) \leq \nu_j^{\text{BM}}(C_{h+1}). \\ \forall j \in \mathcal{J}, a \in X, C \in \mathcal{C} \setminus \{C_1\} : \quad V_j(a, C) \geq \frac{\nu_j^{\text{BM}}(C) - a_j}{M}. \\ \forall j \in \mathcal{J}, a \in X, C \in \mathcal{C} \setminus \{C_1\} : \quad V_j(a, C) \leq \frac{\nu_j^{\text{BM}}(C) - a_j - \varepsilon}{M} + 1. \\ \forall j \in \mathcal{J}, a \in X, C \in \mathcal{C} \setminus \{C_1\} : \quad V_j \geq V_j(a, C). \end{array} \right.$$

PM 2: ICLV

$\sum_{j \in \mathcal{J}} \Psi_j^t(a, l^+(\overline{C}_i))$ soit insuffisant et ne dépasse pas le seuil de majorité, soit qu'au moins un veto opère et empêche une telle affectation.

Pour minimiser le nombre de vetos utilisés, une variable binaire V_j est définie pour chaque critère. La variable vaut 1 ssi un veto est utilisé pour ce critère. $\forall a \in X, C \in \mathcal{C} \setminus \{C_1\}, j \in \mathcal{J}$:

$$V_j \geq V_j(a, C). \quad (\text{veto utilisé})$$

Fonction objectif

Il paraît raisonnable de tenter d'éviter autant que possible l'utilisation de vetos, pour obtenir un modèle aussi simple que possible. La fonction objectif suivante minimise le nombre de critères utilisant un veto.

$$\text{Min} \sum_{j \in \mathcal{J}} V_j.$$

Une alternative pourrait être de minimiser le nombre de situations où un veto intervient.

Le PM 2 synthétise le programme décrit.

3.1.5 CWR : Compute Weights Restriction

Choisir des profils communs induit une contrainte sur les poids qui peuvent être utilisés pour satisfaire les exemples d'affectation d'un décideur $t \in \mathcal{T}$.

3 Convergence de modèles MR Sort pour la décision de groupe

Étant donné un jeu de profils L , et pour un décideur t , notons $W^t(L)$ l'ensemble des coalitions, donc l'ensemble constitué des poids $w_j^t, j \in \mathcal{J}$ et du seuil de majorité λ^t , qui satisfont les exemples d'affectation du décideur.

$$W^t(L) = \{(w_j, j \in \mathcal{J}, \lambda) \mid \forall (a \xrightarrow{t} C_h, C_i) \in E^t : a \succeq C_h \wedge a \not\succeq C_{i+1}\}.$$

Définissons la relation d'importance \triangleright^t sur \mathcal{J} comme suit, $\forall j_1, j_2 \in \mathcal{J}$:

$$j_1 \triangleright^t j_2 \Leftrightarrow w_{j_1} > w_{j_2}, \forall (w_j, j \in \mathcal{J}, \lambda) \in W^t(L).$$

Cette relation indique les comparaisons nécessaires des poids des critères, c'est-à-dire, celles qui sont valides pour toutes les coalitions satisfaisant les exemples du décideur t . La relation \triangleright^t n'est pas nécessairement complète : il peut exister une paire de critères j_1, j_2 telle qu'il existe $(w_j, j \in \mathcal{J}, \lambda) \in W^t(L)$ satisfaisant $w_{j_1} > w_{j_2}$ et $(w'_j, j \in \mathcal{J}, \lambda') \in W^t(L)$ satisfaisant $w'_{j_2} > w'_{j_1}$.

Étant donné un jeu de profils L , le programme CWR calcule \triangleright^t pour chaque décideur $t \in \mathcal{T}$. Il s'agit de chercher, pour chaque paire de critères $j_1, j_2 \in \mathcal{J}$, s'il existe une solution satisfaisant les exemples d'affectation du décideur t , avec une contrainte supplémentaire $w_{j_1}^t \leq w_{j_2}^t$, et avec les profils L . Notons que cette recherche est très rapide, car la connaissance des profils en fait un programme purement linéaire. Si une solution existe, il n'est pas vrai que j_1 est nécessairement plus important que j_2 . Si aucune solution n'existe, cela signifie que $w_{j_1}^t > w_{j_2}^t, \forall (w_j, j \in \mathcal{J}, \lambda) \in W^t(L)$, donc $j_1 \triangleright^t j_2$.

Présenter les relations \triangleright^t aux décideurs pour validation permet de détecter si les profils trouvés à l'étape précédente du processus induisent des contraintes inacceptables pour certains décideurs. Il faut noter que la validation de \triangleright^t par le décideur t ne constitue pas une garantie absolue qu'il accepte l'ensemble des contraintes sur les coalitions imposées par le choix des profils. Cela doit plutôt être considéré comme une approximation raisonnable à ce stade du processus.

Dans le cas où le décideur t ne valide pas \triangleright^t , il existe une paire de critères j_1, j_2 telle que le décideur trouve j_2 au moins aussi important que j_1 alors que \triangleright^t dit l'inverse. Dans ce cas, il est possible de recalculer de nouveaux profils à l'aide de ICL ou ICLV en ajoutant la contrainte $w_{j_2}^t \geq w_{j_1}^t$.

3.1.6 Exemple illustratif

Illustrons les outils proposés sur le scénario hypothétique suivant. Un comité doit choisir un sous-ensemble de projets de recherche à financer, parmi une liste de projets reçus. Le comité désire établir une procédure systématique permettant d'affecter chaque projet à une catégorie parmi trois : les projets excellents, qu'il faut financer (catégorie *Excellent*); les projets de bonne qualité

3.1 Élicitation de paramètres pour méthode MR Sort

Déc 1	Déc 2	Déc 3	Déc 4
Prj 1 → Bon	Prj 31 → Excell.	Prj 61 → Mauv.	Prj 91 → Mauv.
Prj 2 → Excell.	Prj 32 → Mauv.	Prj 62 → Excell.	Prj 92 → Mauv.
Prj 3 → Excell.	Prj 33 → Mauv.	Prj 63 → Mauv.	Prj 93 → Mauv.

TABLE 3.1 – Une partie des exemples d’affectation donnés par les décideurs.

qu’il faudrait financer si un budget supplémentaire peut être trouvé (catégorie *Bon*); les projets de mauvaise qualité qu’il ne faudrait pas financer (catégorie *Mauvais*). Les quatre membres du comité souhaitent utiliser les critères suivants.

- sq** La qualité scientifique du projet, évaluée sur une échelle ordinaire à cinq points.
- wq** La qualité de rédaction, évaluée sur une échelle ordinaire à cinq points.
- ad** L’adéquation entre le projet et les priorités du gouvernement, évaluée sur une échelle ordinaire à trois points.
- te** L’expérience des équipes de recherche associées au projet, évaluée sur une échelle ordinaire à cinq points.
- ic** Une indication binaire indiquant si le projet inclut une collaboration internationale.
- ps** La valeur agrégée des publications des chercheurs participant au projet, évaluée par un nombre entre zéro et cent.

Nous avons conservé les dénominations des critères en anglais de manière à ce qu’elles correspondent aux contenus des fichiers de données publiés sur Internet. Les échelles sont définies de telle manière qu’une plus grande valeur corresponde à une meilleure évaluation.

Chaque décideur a fourni trente exemples d’affectation. Un extrait est indiqué en Table 3.1, et la table 3.2 liste les performances des projets correspondants.

Le programme ICL est utilisé pour trouver des profils partagés par les décideurs qui satisfont leurs exemples individuels. Cependant, avec ces exemples, aucune solution ne peut être trouvée : il n’existe pas de jeux de profils partagés satisfaisant les exemples des quatre décideurs sans véto. Le programme ICLV est alors essayé. Une solution est trouvée, qui reproduit tous les exemples d’affectation avec des profils partagés et un véto sur le critère sq, $\nu_{sq}^{BM}(C_3) = 2$. Les résultats sont donnés en Table 3.3. La table 3.4 indique pour chaque décideur un ensemble de coalitions qui, utilisé avec les profils partagés et le seuil de véto, satisfait leurs exemples d’affectation.

3 Convergence de modèles MR Sort pour la décision de groupe

	sq	wq	ad	te	ic	ps
Prj 1	1	5	1	5	0	72
Prj 2	5	4	1	5	1	6
Prj 3	5	4	3	5	1	50
Prj 31	4	5	3	5	0	98
Prj 32	5	3	1	1	1	65
Prj 33	3	1	2	1	0	53
Prj 61	2	3	2	2	0	62
Prj 62	5	5	1	4	1	67
Prj 63	5	2	2	1	1	9
Prj 91	3	1	1	1	0	89
Prj 92	1	1	2	1	0	12
Prj 93	1	3	3	2	0	40

TABLE 3.2 – Les performances de certaines des alternatives utilisées comme exemples.

Profil	sq	wq	ad	te	ic	ps
$l^-(C_3)$	5	4	3	4	2	27
$l^-(C_2)$	3	4	1	3	0	22
$\nu^{\text{BM}}(C_3)$	2	-	-	-	-	-
$\nu^{\text{BM}}(C_2)$	-	-	-	-	-	-

TABLE 3.3 – Profils et vétos inférés.

Les valeurs trouvées fournissent aux décideurs une interprétation des échelles des critères, ainsi qu'indiqué en section 2.2.3 (voir en particulier la figure 2.3). Ils peuvent alors indiquer si ces valeurs représentent selon eux des frontières adéquates entre les catégories.

Avant de présenter les résultats du calcul d'ICLV aux décideurs, CWR peut être utilisé pour calculer les contraintes sur les poids imposées par le choix des profils et vétos. Ces restrictions sont indiquées en Figure 3.1.

Ce calcul montre qu'un consensus immédiat sur les poids, en utilisant ces profils et vétos et si tous les exemples doivent être satisfaits, est inaccessible. En effet, pour satisfaire les exemples du décideur 3 en utilisant ces profils et vétos, il faut donner un poids plus important au critère ps qu'au critère sq, alors que les exemples des autres décideurs imposent l'exigence inverse.

Constatant ceci, on peut être tenté d'exécuter à nouveau ICLV de manière à trouver des profils et vétos communs qui n'exhibent pas cet inconvénient, donc qui sont tels que les décideurs comparent l'importance de ps et sq de la même façon. Mais ajouter ces contraintes dans ICLV mène à un programme infai-

3.1 Élicitation de paramètres pour méthode MR Sort

Déc	sq	wq	ad	te	ic	ps	λ
Déc 1	0	0,2	0,2	0,4	0	0,2	0,5
Déc 2	0	0,2	0,2	0,4	0	0,2	0,5
Déc 3	0	0,2	0,2	0,3	0,3	0	0,6
Déc 4	0,2	0,2	0,2	0,4	0	0	0,5

TABLE 3.4 – Les jeux de poids et de seuils de majorité trouvés par ICLV. Utiliser ces paramètres avec les profils et vétos inférés reproduit les exemples des décideurs correspondants.

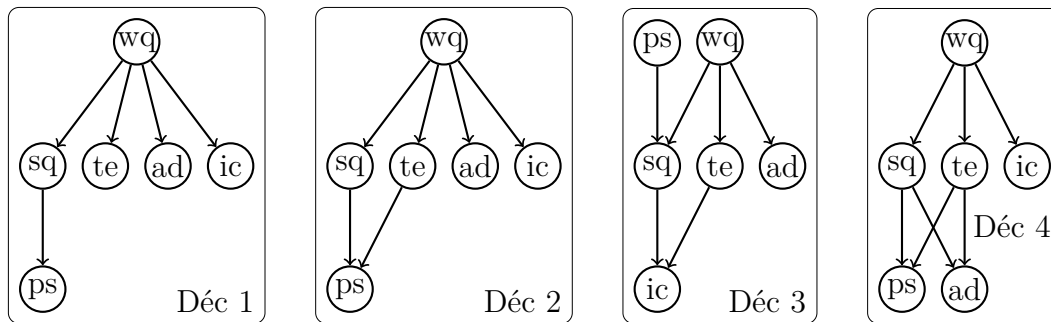


FIGURE 3.1 – Les restrictions sur les poids imposés par le choix de profils et vétos communs, pour chaque décideur. Une flèche d'un critère j à un critère j' indique que le choix de ces profils et la satisfaction des exemples donnés par le décideur requièrent un poids strictement plus grand pour j que pour j' pour ce décideur.

sable : il n'existe pas de profils et vétos communs satisfaisant tous les exemples d'affectation en utilisant des poids individuels satisfaisant la contrainte additionnelle. Alternativement, le programme IMRS aurait pu être utilisé, mais le résultat aurait bien sûr été négatif.

Les résultats trouvés précédemment, d'ICLV et de CWR, sont alors montrés aux décideurs, qui les valident. Cette acceptation indique que les décideurs trouvent que les partitions des échelles des critères induites par les profils et le veto représentent de manière acceptable les limites des catégories. Ils savent par ailleurs que pour parvenir à un modèle de préférence commun, certains d'entre eux devront accepter de revoir certains de leurs exemples d'affectation. La méthode proposée par Damart *et al.* (2007) peut être utilisée pour construire itérativement un consensus sur les poids parmi le groupe de décideurs. Nous ne détaillons pas ce processus ici. En supposant que le décideur 3 change d'avis concernant le projet 90 et 69, et que le décideur 4 change d'avis concernant

3 Convergence de modèles MR Sort pour la décision de groupe

	sq	wq	ad	te	ic	ps
$l^+(C_2)$	5	4	3	4	2	27
$l^+(C_1)$	3	4	1	3	0	22
$\nu^{\text{BM}}(C_3)$	2	-	-	-	-	-
$\nu^{\text{BM}}(C_2)$	-	-	-	-	-	-
poïds	0,09	0,18	0,18	0,37	0,09	0,09
$\lambda = 0,5$						

TABLE 3.5 – Le modèle de préférence final.

le projet 98, un modèle de préférence consensuel peut représenter tous les exemples. Il est indiqué en Table 3.5.

3.1.7 Performance des algorithmes

Les algorithmes ont été implémentés dans la bibliothèque Java J-MCDA (cf. Section 6.2) et utilisent la bibliothèque JLP pour définir les PMs. Les PMs ont ensuite été résolus à l'aide du programme Ilog CPLEX, version 12.2. La couche Java a permis de faciliter l'intégration des algorithmes dans le projet Decision Deck. L'impact en termes de performances est négligeable car les PMs sont résolus en C grâce à la technologie Ilog concert. Les tests ont été effectués sur un PC Intel Core2 Duo E8400 3 GHz avec 2 GB RAM, sous Debian Lenny, un système d'exploitation GNU/Linux.

Nous avons testé les performances des programmes ICL et CWR sur environ 2000 problèmes, générés aléatoirement, de tailles diverses prises dans les limites suivantes : de trois à dix critères, de un à quatre décideurs, de deux à cinq catégories (donc de un à quatre profils), avec entre 1 et 700 exemples donnés par chaque décideur. Les critères ont tous une échelle entière entre 0 et 99.

Une fois la taille choisie, nous générons des profils représentant le mode de raisonnement des décideurs fictifs. Les profils répartissent l'espace des alternatives également entre les catégories. Par exemple avec quatre catégories, les profils ont une performance sur chaque critère de 25 pour le plus mauvais profil, 50 pour le profil du milieu, et 75 pour le meilleur profil. Chaque décideur est également associé à un choix aléatoire de coalitions, soit un ensemble de poids et un seuil de coupe. Ces coalitions, associées aux profils communs, représentent un décideur fictif, que nous supposons ici parfaitement cohérent. Pour occuper nos décideurs fictifs, nous générons des alternatives aléatoirement : chaque alternative a , sur chaque critère, une performance tirée selon une distribution uniforme sur l'échelle de 0 à 99. Les décideurs fictifs rangent docilement chaque alternative dans la catégorie indiquée par leur modèle de

3.1 Élicitation de paramètres pour méthode MR Sort

Variables binaires	Taille de l'échantillon	Résolus
[0, 399]	477	100%
[400, 799]	441	87%
[800, 1199]	362	80%
[1200, 1599]	290	78%
[1600, 1999]	268	75%
[2000, 2199]	121	69%

TABLE 3.6 – La proportion de problèmes résolus en moins de 90 minutes et en utilisant moins de 3 GB d'espace disque.

préférence, ce qui fournit les exemples d'affectation qui seront donnés au programme ICL. Les exemples d'affectation sont uniquement constitués d'intervalles d'une seule catégorie, et les décideurs partagent toutes les alternatives utilisées comme exemples dans ce test expérimental. Mentionnons également que la variante utilisée pour ICL comprenait l'objectif de maximisation du slack s de la section 3.1.3 mais pas les optimisations qui seront présentées au chapitre 8.

Performance

Observons la proportion de problèmes pour lesquels un ensemble de profils a été trouvé par le programme ICL en moins de 90 minutes¹ et avec une limite d'utilisation d'espace disque de 3 GB, représentant la taille de la structure utilisée par le solveur. La limite de temps a été fixée à 90 minutes car cela semble être une limite de temps raisonnable pour l'usage que nous envisageons (nous y reviendrons). La proportion de problèmes qui peuvent être résolus dans ces limites dépend bien sûr de leurs tailles. La table 3.6 indique la proportion de problèmes résolus en fonction du nombre de variables binaires dans le programme. La taille de l'échantillon indique le nombre d'instances générées, soit le dénominateur de la proportion calculée. Les mêmes tendances sont observées en considérant le nombre de variables continues au lieu du nombre de variables binaires.

Le nombre de variables binaires dans ICL égale le nombre de critères multiplié par le nombre de profils à trouver, que multiplie le nombre d'alternatives dans l'ensemble d'exemples X . Notons que ce dernier nombre n'est pas nécessairement l'addition du nombre d'exemples donnés par chaque décideur, car la même alternative peut être utilisée comme exemple par plusieurs décideurs.

1. Voir la note concernant le (més)usage de la langue. Le lecteur notera que nous avons insidieusement juxtaposé deux difficultés dans cette phrase.

3 Convergence de modèles MR Sort pour la décision de groupe

Nb de critères	Nb de variables binaires	Taille de l'échantillon	Résolus
[3, 5]	[0, 2199]	714	100%
[6, 7]	[0, 399]	112	100%
[6, 7]	[400, 799]	108	96%
[6, 7]	[800, 1199]	93	96%
[6, 7]	[1200, 1599]	84	96%
[6, 7]	[1600, 1999]	80	95%
[6, 7]	[2000, 2199]	24	96%
[8, 10]	[0, 399]	184	99%
[8, 10]	[400, 799]	179	70%
[8, 10]	[800, 1199]	132	48%
[8, 10]	[1200, 1599]	100	39%
[8, 10]	[1600, 1999]	97	36%
[8, 10]	[2000, 2199]	52	29%

TABLE 3.7 – La proportion de problèmes résolus en moins de 90 minutes et en utilisant moins de 3 GB d'espace disque, en fonction du nombre de critères et de variables binaires contenus dans le problème.

Un problème réel où les exemples sont fournis par les décideurs manuellement contiendra probablement moins de 1200 variables binaires, ce nombre étant atteint pour un problème contenant six critères, trois catégories, cent alternatives utilisées comme exemples. Une nette majorité de problèmes de cette taille peuvent être résolus en 90 minutes. Cette durée semble raisonnable si le programme est utilisé par l'analyste entre des réunions avec les décideurs. Par contre, les performances actuelles ne permettent pas de traiter une situation où les exemples seraient tirés en grand nombre depuis une base de données de cas passés, ou si une utilisation interactive du programme est souhaitée.

La proportion de problèmes résolus dépend grandement, même en neutralisant l'effet du nombre de variables, de la structure du problème, et en particulier du nombre de critères. En effet, tous les problèmes de notre expérience comptant moins de six critères ont été résolus, même ceux contenant plus de 2000 variables binaires. Au contraire, quand le problème contient huit à dix critères, moins de 50% des instances impliquant 1200 variables binaires peuvent être résolues en 90 minutes. La table 3.7 montre la proportion de problèmes résolus en fonction du nombre de critères et de variables binaires impliqués.

Les performances de CWR ont été mesurées également. L'algorithme a été lancé sur chaque instance testée pour ICL ayant abouti à une solution. L'algorithme n'a jamais eu besoin de plus de quelques minutes pour calculer toutes les restrictions, donc l'entièreté de la relation \triangleright^t , pour tous les décideurs im-

3.1 Élicitation de paramètres pour méthode MR Sort

pliés. Dans la plupart des cas, quelques secondes suffisent. En effet, le calcul de ces restrictions requiert de résoudre un grand nombre de PLs, mais chacun d'eux se résoud très rapidement.

Ces données montrent que le programme ICL résoud la plupart des problèmes ayant une taille raisonnable en 90 minutes. Dans l'approche que nous proposons, ce programme sera utilisé pour analyser les préférences indiquées par les décideurs dans l'intervalle entre deux réunions. Une limite temporelle de 90 minutes pourrait donc être considérée comme raisonnable. Repousser cette limite temporelle pourrait peut-être permettre de résoudre de plus gros problèmes. Cependant, il serait probablement plus approprié de développer d'autres méthodes de résolution plus performantes tenant compte de la structure du problème pour s'attaquer à des problèmes de plus grande taille ou permettre une utilisation interactive.

Comportement de l'algorithme

Il serait utile d'avoir une idée du nombre d'exemples nécessaires à la capture d'un modèle de préférence adéquat d'un ou plusieurs décideurs. Cette section présente un test expérimental en vue d'apporter un élément de réponse à cette question.

Considérons les problèmes pour lesquels ICL a trouvé une réponse, donc un jeu de profils communs, dans l'expérience décrite ci-dessus. Nous pouvons comparer les profils trouvés aux profils des décideurs fictifs, c'est-à-dire aux profils originaux utilisés pour générer les affectations. Le jeu de profils original est noté $L \equiv \{l_j(C), \forall C \in \mathcal{C} \setminus C_1, j \in \mathcal{J}\}$, le jeu de profils trouvé par l'algorithme est noté $L' \equiv \{l'_j(C), \forall C \in \mathcal{C} \setminus C_1, j \in \mathcal{J}\}$. La notation $l_j(C)$ désigne le profil inférieur correspondant à la catégorie C , précédemment noté $l_j^-(C)$. Nous omettons ici l'exposant pour simplifier les notations.

Définissons une mesure de similarité $s_\varepsilon(L, L')$ pour comparer les deux jeux de profils L, L' , où ε est un paramètre de tolérance. Cette mesure considère la proportion de performances de profils égaux, à ε près, dans les deux jeux de profils. Définissons le prédicat logique $e_\varepsilon(l_j(C), l'_j(C))$ indiquant, pour un critère j et une catégorie C donnés, si les deux performances $l_j(C), l'_j(C)$ sont égales à ε près :

$$e_\varepsilon(l_j(C), l'_j(C)) \Leftrightarrow |[l_j(C)] - [l'_j(C)]| \leq \varepsilon.$$

Les performances sont arrondies vers le haut avant d'être comparées, car les échelles des critères sont entières. Une performance de 3,7 est donc sémantiquement identique à une performance de 4 ou de 3,1. La mesure de similarité

3 Convergence de modèles MR Sort pour la décision de groupe

Nb d'exemples	Nb pbl	0	1	2	3	4	5	10
[0, 79]	662	4%	11%	16%	20%	24%	27%	41%
[80, 159]	311	15%	30%	39%	47%	54%	58%	73%
[160, 239]	214	26%	44%	55%	62%	66%	71%	84%
[240, 319]	118	31%	53%	66%	75%	80%	84%	92%
[320, 399]	69	43%	63%	74%	79%	84%	87%	92%
[400, 479]	66	45%	68%	81%	87%	91%	92%	96%
[480, 559]	62	53%	76%	87%	91%	96%	97%	99%
[560, 639]	20	52%	86%	92%	96%	99%	100%	100%
[640, 719]	27	66%	84%	93%	96%	96%	97%	98%
[720, 799]	27	64%	83%	91%	94%	95%	97%	98%
[800, 879]	15	70%	97%	99%	100%	100%	100%	100%

TABLE 3.8 – La similarité moyenne entre les profils trouvés et les originaux, par intervalle du nombre total d'exemples dans le problème, pour ε égal à zéro à cinq et à dix.

est définie comme suit.

$$s_\varepsilon(L, L') = \frac{|\{(j, C) \in \mathcal{J} \times \mathcal{C} \setminus C_1 \mid e_\varepsilon(l_j(C), l'_j(C))\}|}{|\mathcal{J}| |\mathcal{C} \setminus C_1|}.$$

Par exemple, dans un problème à trois catégories $\{C_1, C_2, C_3\}$ et trois critères, si les deux profils originaux ont des valeurs sur les trois critères $l(C_2) = \langle 30; 30; 30 \rangle$ et $l(C_3) = \langle 60; 60; 60 \rangle$, et que le programme ICL trouve une solution $l'(C_2) = \langle 31, 7; 32, 1; 29, 2 \rangle$ et $l'(C_3) = \langle 60, 2; 57, 2; 50, 1 \rangle$, alors la similarité avec une tolérance $\varepsilon = 2$ entre les deux jeux de profils vaudra $s_2(L, L') = 4/6 = 67\%$.

Il est évident que la similarité moyenne entre les profils inférés et les profils originaux augmente avec le nombre d'exemples d'affectation fournis à ICL. Pour un intervalle de nombre d'exemples donnés, nous avons considéré l'ensemble P de tous les problèmes ayant un nombre d'exemples dans cet intervalle, calculé pour chaque problème de cet ensemble la similarité, et divisé la somme des similarités par le nombre de problèmes dans P pour obtenir une similarité moyenne. Les tables 3.8 et 3.9 montrent les résultats. La deuxième colonne indique la taille de l'échantillon, à savoir le nombre de problèmes considérés dans le calcul de la moyenne, les colonnes suivantes indiquent les mesures de similarité pour des valeurs de ε croissantes.

Notons que la tendance observée ici est très semblable lorsque seuls les petits problèmes sont considérés (contenant peu de profils et de critères), ou lorsque de gros problèmes sont considérés, ce qui semble indiquer que l'impact du

3.1 Élicitation de paramètres pour méthode MR Sort

Nb ex par décideur	Nb pbl	0	1	2	3	4	5	10
[0, 39]	620	4%	10%	16%	20%	24%	27%	42%
[40, 79]	390	15%	30%	40%	48%	54%	58%	72%
[80, 119]	227	28%	47%	59%	67%	72%	76%	86%
[120, 159]	132	37%	58%	70%	76%	80%	83%	90%
[160, 199]	77	47%	66%	73%	77%	79%	82%	88%
[200, 239]	66	52%	75%	83%	89%	91%	92%	95%
[240, 279]	54	61%	84%	92%	94%	96%	97%	98%
[280, 319]	27	64%	93%	95%	97%	98%	99%	99%
[320, 359]	22	68%	82%	85%	86%	88%	90%	94%
[360, 399]	12	74%	89%	95%	95%	96%	98%	98%

TABLE 3.9 – La similarité moyenne entre les profils trouvés et les originaux, par intervalle du nombre d'exemples dans le problème par décideur, pour ε égal à zéro à cinq et à dix.

nombre de paramètres que le PM doit trouver est correctement neutralisé par le dénominateur de la mesure de similarité, égal au nombre de paramètres à trouver.

Ces données montrent qu'un grand nombre d'exemples d'affectation est requis pour que ICL trouve des valeurs de performance proches des valeurs originales. Par exemple, supposons que la tolérance acceptée soit de $\varepsilon = 5$, donc une tolérance de cinq pourcents sur l'échelle de critères à cent points utilisée ici, et que l'on souhaite atteindre une similarité de 70%, donc au moins sept dixième des performances des profils doit être proche des valeurs que les décideurs sont supposées avoir en tête. Alors, d'après les problèmes testés dans cette expérience, il faut demander au total 160 à 240 exemples d'affectation au groupe de décideurs.

Ce résultat pourrait être amélioré en choisissant les exemples d'affectation soumis aux décideurs. Dans notre expérience, les performances des alternatives servant d'exemples sont déterminées aléatoirement. Cela peut conduire à une information ajoutée très pauvre, en moyenne pour chaque exemple d'affectation, en termes de réduction de l'espace des modèles de préférence compatibles. Une piste de recherche intéressante est de développer des algorithmes d'apprentissage actifs (nous empruntons le terme « *active learning* » utilisé en machine learning (Fürnkranz et Hüllermeier, 2011)), capables de choisir, à une certaine étape du processus, quels exemples les décideurs devraient fournir en plus pour guider l'apprentissage de la manière la plus efficace possible.

Cependant, il faut garder à l'esprit que la méthode proposée ici sert à mettre d'accord un groupe de décideurs, et non à retrouver un ensemble de profils

3 Convergence de modèles MR Sort pour la décision de groupe

supposés pré-existants dans leur tête. Ainsi, les résultats de cette expérience éclairent à notre avis surtout la question de la généralité du modèle MR Sort. Ils ne devraient pas être considérés comme un bon indicateur de performance d'une méthode de décision de groupe.

L'étude menée par Leroy *et al.* (2011), détaillée à la section 2.3.3, s'intéresse également à la question du nombre d'exemples requis pour déterminer un modèle MR Sort, mais l'aborde sous un autre angle : les auteurs infèrent un modèle complet, pour un seul décideur. Ils constatent qu'il faut, par exemple, 70 exemples d'affectation pour obtenir un modèle qui commet moins de 10% d'erreurs d'affectation par rapport au modèle original, dans une situation impliquant trois catégories et cinq critères. Cette constatation est convergente avec nos résultats qui indiquent qu'un grand nombre d'exemples est requis pour retrouver les valeurs de profils originales.

3.1.8 Conclusions et perspectives

Nous avons présenté une méthode d'élicitation de préférences pour un groupe de décideurs. Elle vise à obtenir un modèle de type MR-Sort. Elle s'inscrit donc dans l'approche visant à obtenir un modèle précis. Ceci peut nécessiter des choix arbitraires à certaines étapes. Par exemple, le PM ICL exhibe une solution satisfaisante, et non l'ensemble des solutions compatibles avec les exemples. Comme indiqué en section 1.3.2, l'objectif d'aboutir à un modèle précis est approprié si le groupe souhaite établir un standard définissant des affectations non ambiguës.

Notre proposition contient deux apports bien distincts, l'un technique, l'autre méthodologique. D'une part, nous proposons des PMs pour représenter le problème d'inférence de modèles compatibles comme des PLNEs. Les tests de performance montrent que le PM ICL peut inférer un modèle en un temps raisonnable pour des problèmes contenant moins de huit critères.

L'apport méthodologique consiste à proposer l'inférence d'une partie du modèle, les profils, de manière à découper le problème de l'établissement du modèle en deux phases. Pour la deuxième phase, partant des profils acceptés, nous proposons d'utiliser l'approche de Damart *et al.* (2007) (décrite en section 2.3.5) pour établir les coalitions.

Cet aspect de notre contribution ne s'applique pas qu'au modèle MR Sort. L'élicitation d'autres types de modèles pourrait être découpée en deux phases. L'approche s'applique cependant particulièrement bien lorsque le modèle requiert l'élicitation de paramètres de natures différentes, tels que les profils et les coalitions. L'approche pourrait également, avec MR Sort, être utilisée dans l'autre sens, pour trouver d'abord des poids partagés en laissant les profils varier. Dans notre présentation, nous cherchons d'abord des profils consensuels

3.1 Élicitation de paramètres pour méthode MR Sort

car cela détermine une compréhension partagée des échelles X_j des critères. Cela pourrait donner une base plus stable au processus d'élicitation que le procédé inverse.

Perspectives

Les instances comprenant plus de huit critères posent des problèmes computationnels difficiles. Même les problèmes de plus petite taille peuvent requérir un temps de résolution incompatible avec une utilisation interactive des outils. Des algorithmes de résolution spécifiques devraient donc être étudiés.

Il serait intéressant d'adapter le PM ICL de manière à sélectionner un jeu de profils qui minimise les restrictions telles que calculées par CWR ou par un autre indicateur similaire.

Lorsque les exemples d'affectation ne sont pas compatibles avec un modèle MR Sort utilisant des profils partagés, des approches existantes peuvent être facilement adaptées pour trouver les exemples à modifier de manière à pouvoir trouver des profils communs (Mousseau *et al.*, 2006, 2003). Dans un tel cas, il serait également intéressant de pouvoir trouver facilement les sous-ensembles de décideurs qui semblent plus proches d'un consensus. Cet objectif a été poursuivi dans l'application présentée en section 3.3, mais la recherche de sous-ensembles de décideurs proches a été effectuée sans aide d'outils adaptés. De plus, d'autres mesures pourraient être utilisées pour capturer la notion de proximité de raisonnement entre deux décideurs.

Une question importante d'un point de vue pratique concerne le choix des alternatives considérées dans les exemples d'affectation. Il serait intéressant de trouver une stratégie pour bien les choisir, de façon à amener le plus d'informations possibles.

Une question qui nous semble particulièrement intéressante est liée au problème du surajustement mentionné brièvement lors de la description de ICLV : si l'on tolère l'utilisation de nombreux vétos, il sera peut-être très souvent possible de trouver un modèle satisfaisant les exemples fournis par les décideurs, tant la classe de modèles dans laquelle le modèle est pris deviendra générale. Mais il serait dans ce cas difficile de prétendre que le modèle trouvé représente effectivement le décideur. Cette constatation n'est bien sûr pas spécifique à la classe de modèles MR Sort. Pour étudier plus avant cette question, il nous semble nécessaire de d'abord mieux définir quelle sorte de préférences les modèles utilisés en AMCD prétendent représenter. Bien qu'il s'agisse là d'une question de long terme, nous proposons un début de réflexion ce concernant à la section 9.2.

Comparaison et validation

Notre proposition pourrait être comparée à une autre approche possible consistant à utiliser uniquement la méthode de Damart *et al.* (2007) pour établir l'ensemble du modèle, en procédant donc en une seule phase. L'avantage de notre proposition est qu'elle pourrait permettre une progression plus facile : fixer les profils réduit fortement l'espace des paramètres dans lequel il faut encore chercher un consensus. L'inconvénient est qu'il se pourrait que le choix initial d'un jeu de profils emmène sur une voie sur laquelle il n'est pas possible de trouver un consensus, alors qu'un consensus aurait pu être trouvé avec un autre jeu de profils. Nous proposons un outil de mesure, CWR, pour réduire ce risque, mais il ne peut pas être évité totalement.

Des expériences devraient être menées pour comparer différentes approches. Des mesures pourraient être utilisées pour quantifier la difficulté de parvenir à un consensus, telles que le temps de discussion total requis, le nombre d'exemples fournis, la satisfaction des décideurs à l'issue du processus. Il serait intéressant également d'évaluer l'ampleur du changement d'avis des décideurs durant le processus. Peut-être arriverait-il que certains décideurs soient, après discussion avec le groupe, convaincus que certains de leurs exemples d'affectation fournis au début n'étaient pas affectés à la catégorie appropriée. Notons bien que mesurer le changement d'avis ne requiert pas de se prononcer sur l'opportunité de changer d'avis. Juger si un tel changement d'avis serait positif est une question de recherche en soi.

3.2 Calcul de similarité entre deux modèles MR Sort

Lors de tests expérimentaux tels que celui décrit ci-dessus, il serait pratique de pouvoir comparer deux modèles de préférence du point de vue de leur résultat. Nous avons comparé directement les valeurs des paramètres, mais ce n'est pas parfaitement satisfaisant. Deux modèles pourraient avoir des valeurs de paramètres très différentes et pourtant induire des fonctions de tri quasiment, voire parfaitement, identiques. Il serait intéressant de connaître, étant donné une distribution de probabilité postulée sur les performances possibles des alternatives, la probabilité que les deux fonctions de tri comparées affectent une alternative différemment. Cette question peut être posée de manière générale, indépendamment d'une méthode de tri particulière. Par exemple, on peut vouloir comparer probabilistiquement deux modèles de préférence pour une méthode basée sur la théorie de la valeur. On peut même envisager de comparer probabilistiquement deux modèles de préférence basés sur des méthodes

différentes. Une manière simple d'obtenir une approximation de la probabilité d'affecter une alternative différemment selon deux modèles de préférence est d'utiliser une simulation aléatoire : à chaque étape, il suffit de tirer aléatoirement une alternative dans l'espace des alternatives possibles, suivant la distribution de probabilité que nous avons supposé connue, de la classer avec les deux fonctions de tri, et d'observer si le résultat est différent. Après un grand nombre de tirages, on devrait avoir une bonne idée de la probabilité de désaccord des deux fonctions. Cette technique a été utilisée dans le cas de la comparaison de deux modèles MR Sort sans véto par Leroy *et al.* (2011). L'inconvénient de cette approche est qu'elle n'atteint la vraie valeur qu'asymptotiquement.

Dans cette section, nous proposons une méthode permettant de calculer exactement la probabilité que deux modèles de préférence MR Sort sans véto affectent différemment une alternative tirée au hasard.

Soit un espace $X = \prod_{j \in \mathcal{J}} X_j$, espace des alternatives possibles ; soit \mathcal{C} l'ensemble des catégories auxquelles chaque modèle peut affecter une alternative. Nous supposons bien sûr que les deux modèles partagent le même ensemble de catégories, sinon le problème n'est pas bien défini. Soit $|\mathcal{C}| = k$ le nombre de catégories, donc chaque modèle contient $k - 1$ profils. Nous supposons connue la distribution de probabilité sur X , et l'ordre de préférence sur chaque critère. Soient deux modèles de préférence MR Sort ω_1, ω_2 , contenant chacun un ensemble de profils $L^\omega \subseteq X$, de poids $\{w_j^\omega, j \in \mathcal{J}\}$, et un seuil de majorité λ^ω , $\omega \in \{\omega_1, \omega_2\}$. Les performances des profils sont notées $l_j^\omega, j \in \mathcal{J}$. Supposons également que chaque échelle X_j admette un minimum, noté \underline{X}_j , et un maximum, noté \overline{X}_j . Ce n'est pas une hypothèse restrictive, car l'algorithme peut très simplement être modifié pour admettre des ensembles X_j non bornés, mais cela simplifie la description de l'algorithme.

3.2.1 En considérant toutes les catégories

Pour chaque critère j , considérons l'ensemble de toutes les valeurs de profils $E_j = \{l_j^{\omega_1}, \forall l \in L^{\omega_1}\} \cup \{l_j^{\omega_2}, \forall l \in L^{\omega_2}\} \cup \underline{X}_j \cup \overline{X}_j$. Ordonnons ces valeurs en utilisant la relation de préférence définie sur le critère j . Nous obtenons, pour un critère donné, une liste de maximum $2(k - 1) + 2 = 2k$ valeurs, chacune dans X_j . Construisons un ensemble d'intervalles $I_j = \{[i_j^1, i_j^2[, i_j^1, i_j^2 \in E_j\}$, représentant les valeurs qu'une alternative peut prendre, en ne considérant que les valeurs consécutives dans la liste E_j ordonnée. Il y a maximum $2k - 1$ tels intervalles, pour un critère donné. Prenons le produit cartésien, sur chaque critère, de cet ensemble d'intervalles, $\mathcal{I} = \prod_{j \in \mathcal{J}} I_j$. Notons que $|\mathcal{I}| \leq (2k - 1)^{|\mathcal{J}|}$.

Un élément $I \in \mathcal{I}, I \subseteq X$, où $I = [i_{j_1}^1, i_{j_1}^2[\times [i_{j_2}^1, i_{j_2}^2[\times \dots$, représente un sous-

3 Convergence de modèles MR Sort pour la décision de groupe

ensemble particulier de l'espace des alternatives possibles. Il est particulier en ce sens que toute alternative qui se trouve dans ce sous-ensemble sera affectée à la même catégorie par une fonction de tri donnée f_ω . Il suffit donc de considérer n'importe quel point $a \in I$, par exemple $a = \langle i_{j_1}^1, i_{j_2}^1, \dots \rangle$, et de calculer son affectation C_I^ω à l'aide des deux fonctions de tri $f_\omega, \omega \in \{\omega_1, \omega_2\}$. La probabilité $d(I)$ qu'une alternative soit dans l'intervalle I peut être calculée à partir de la distribution. Les alternatives $a \in I$ sont affectées de la même manière par les deux modèles ssi $C_I^{\omega_1} = C_I^{\omega_2}$. La probabilité cherchée est donc :

$$P_{\text{mêmes affectations}} = \sum_{\{I \in \mathcal{I} \mid C_I^{\omega_1} = C_I^{\omega_2}\}} d(I).$$

La complexité au pire cas de l'algorithme de calcul de la probabilité est de l'ordre de $|\mathcal{J}| (2k-1)^{|\mathcal{J}|}$, car la complexité du calcul d'affectation est de l'ordre de $|\mathcal{J}|$. Nous supposons que le calcul de $d(I)$ ne dépasse pas cette complexité. Pour $k = 5, |\mathcal{J}| = 10$, ce nombre égale 3×10^{10} .

L'algorithme peut être accéléré en pratique, bien que cela ne réduise pas sa complexité au pire cas, de la façon suivante. L'algorithme présenté sélectionne une alternative dans chaque intervalle I possible et calcule l'affectation résultante. Mais il est possible que le choix de performances $a_j, j \in \mathcal{J}'$ sur un sous-ensemble de critères $\mathcal{J}' \subset \mathcal{J}$ suffise à déterminer univoquement l'affectation de l'alternative selon f_{ω_1} et f_{ω_2} . Dès que la catégorie d'affectation peut être calculée sans ambiguïté, il n'est pas nécessaire de continuer à fixer les performances sur les autres critères. Il suffit, si les affectations sont égales selon les deux fonctions de tri, de comptabiliser en faveur la probabilité $d(\prod_{j' \in \mathcal{J}'} I_{j'} \times \prod_{j \in \mathcal{J} \setminus \mathcal{J}'} X_j)$. Que les affectations soient identiques ou non, l'algorithme peut ensuite passer outre les cas qu'il aurait autrement dû considérer en choisissant des intervalles parmi les critères dans $\mathcal{J} \setminus \mathcal{J}'$.

Il est sans-doute intéressant, pour implémenter cette optimisation, de considérer d'abord les critères de plus grand poids, afin d'augmenter les chances de déterminer univoquement une catégorie d'affectation rapidement. Plus précisément, comme les modèles doivent chacun pouvoir déterminer l'affectation, nous proposons d'ordonner les critères par ordre décroissant du minimum des poids selon les deux modèles, $\min_{\omega \in \{\omega_1, \omega_2\}} w_j^\omega$.

L'algorithme proposé permet également facilement de donner des réponses intermédiaires à l'utilisateur. Cela peut être utile si les modèles à comparer contiennent de nombreux paramètres, entraînant dans ce cas un temps de calcul total éventuellement important. L'algorithme garde en mémoire, au fur et à mesure de sa progression dans les intervalles $I \in \mathcal{I}$, la somme des probabilités $P_{\text{borne inférieure}}^{\mathcal{I}'} = \sum_{\{I \in \mathcal{I}' \mid C_I^{\omega_1} = C_I^{\omega_2}\}} d(I)$ indiquant une similarité. À tout moment, en notant $\mathcal{I}' \subseteq \mathcal{I}$ l'ensemble d'intervalles explorés jusqu'alors, cette

valeur représente une borne inférieure de la probabilité cherchée. La valeur $P_{\text{borne inférieure}}^{\mathcal{I}'}$ s'approche de la vraie valeur $P_{\text{mêmes affectations}}$ au fur et à mesure que l'algorithme explore \mathcal{I} . L'algorithme peut également calculer la somme des probabilités correspondant aux intervalles pour lesquels les deux modèles divergent : $P_{\text{affectations différentes}}^{\mathcal{I}'} = \sum_{\{I \in \mathcal{I} | C_I^{\omega_1} \neq C_I^{\omega_2}\}} d(I)$, et conserver en mémoire $P_{\text{borne supérieure}}^{\mathcal{I}'} = 1 - P_{\text{affectations différentes}}^{\mathcal{I}'}$. La différence entre $P_{\text{borne inférieure}}^{\mathcal{I}'}$ et $P_{\text{borne supérieure}}^{\mathcal{I}'}$ s'amenuisera au fur et à mesure de la progression de l'algorithme.

3.2.2 Par paires de catégories

Une stratégie (proposée oralement par Marc Piriot) permet d'utiliser l'algorithme proposé ci-dessus tout en évitant la complexité $\theta(|\mathcal{J}|(2k-1)^{|\mathcal{J}|})$.

Notons $P(\geq C_i, \geq C_j)$ la probabilité pour une alternative d'être affectée à une catégorie au moins aussi bonne que C_i d'après le premier modèle de préférence et d'être affectée à une catégorie au moins aussi bonne que C_j d'après le second modèle de préférence. Cette valeur peut être calculée par une variante de l'algorithme vu précédemment, expliquée ci-dessous. Montrons d'abord que la connaissance de certaines de ces valeurs suffit à déterminer la probabilité cherchée.

Notons $P(C_i, C_j)$ la probabilité qu'une alternative soit affectée exactement à la catégorie C_i d'après le modèle ω_1 et exactement à la catégorie C_j d'après le deuxième. Les notations $P(\geq C_i, C_j)$ et $P(C_i, \geq C_j)$ sont définies de manière similaire.

L'égalité suivante permet de déterminer la probabilité cherchée.

$$\begin{aligned}
 P &= \sum_{i \in [1, k]} P(C_i, C_i) \\
 &= \sum_{i \in [1, k-1]} \{P(C_i, C_i)\} + P(C_k, C_k) \\
 &= \sum_{i \in [1, k-1]} \{[P(\geq C_i, C_i)] - [P(\geq C_{i+1}, C_i)]\} \\
 &\quad + P(\geq C_k, \geq C_k) \\
 &= \sum_{i \in [1, k-1]} \{[P(\geq C_i, \geq C_i) - P(\geq C_i, \geq C_{i+1})] \\
 &\quad - [P(\geq C_{i+1}, \geq C_i) - P(\geq C_{i+1}, \geq C_{i+1})]\} \\
 &\quad + P(\geq C_k, \geq C_k).
 \end{aligned}$$

Il suffit donc de calculer les $3k + 1$ valeurs de $P(., .)$ requises. Pour ce faire, l'algorithme précédent peut être utilisé, en considérant seulement un profil

pour chaque modèle. Pour calculer $P(C_i, C_j)$, il faut seulement distinguer les affectations de f_{ω_1} à une catégorie moins bonne que C_i et les affectations à au moins C_i , donc il suffit de considérer le profil $l^-(C_i)$. Le même raisonnement s'applique au choix du profil concernant ω_2 . La complexité de l'algorithme pour le calcul d'une valeur de $P(., .)$ est donc réduite à $|\mathcal{J}|(2 \times 2 - 1)^{|\mathcal{J}|} = |\mathcal{J}|3^{|\mathcal{J}|}$. Cela donne une complexité totale pour calculer la similarité entre deux modèles avec cette stratégie de l'ordre de $k|\mathcal{J}|3^{|\mathcal{J}|}$. Pour $k = 5, |\mathcal{J}| = 10$, ce nombre vaut 3×10^6 . Il faut noter qu'avec deux catégories, l'optimisation proposée en fin de section 3.2.1 aura probablement un effet important. Cela nous conduit à penser que l'algorithme proposé pourra répondre rapidement dans un grand nombre de situations impliquant des modèles de taille raisonnable.

L'algorithme a été implémenté et est disponible dans la bibliothèque J-MCDA (cf. Section 6.2). Des tests aléatoires permettant d'évaluer sa performance et de tester les différentes stratégies d'optimisation proposées sont prévus comme une piste de recherche future. Il serait également intéressant d'imaginer des algorithmes capables de comparer des modèles de préférence pour d'autres méthodes de tri telles que les fonctions de valeur.

3.3 Application réelle : échelle de risque pour produits polluants

Cette section décrit un processus d'AMCD où nous avons utilisé le PM ICL pour obtenir des modèles de préférence de groupes. Le processus suggéré à la section 3.1.2 n'a pas été suivi ici : les outils ont été mobilisés d'une manière très différente. Il y a bien sûr de nombreuses manières de combiner ces outils proposés dans ce chapitre et les outils existants de façon à aider un groupe de décideurs. L'objectif de cette section est de montrer une autre combinaison possible.

Les données correspondant à cette application sont disponibles en format électronique (voir annexe D, numéro 2).

3.3.1 Contexte

Nous avons été consultés par une équipe d'Euréal, « Centre Européen d'Expertise et d'Evaluation », pour les aider à analyser les préférences exprimées par un groupe de participants dans une étude concernant les risques liés à l'utilisation de produits polluants.

L'équipe d'Euréal avait elle-même été mandatée pour consulter un échantillon de personnes visant à obtenir des informations de préférence de leur part sur ce sujet.

3.3 Application réelle : échelle de risque pour produits polluants

L'objectif du commanditaire est d'établir une méthode de rangement par priorité de traitement de produits polluants : étant donné ses caractéristiques sur un certain nombre de critères, quelle priorité de traitement devrait lui être accordée ? Un traitement est ici une action visant à réduire son potentiel impact nocif sur la population ou l'environnement, ou une action d'étude supplémentaire, une proposition de législation, etc.

Notre intervention a pour but de proposer des modèles permettant de représenter au mieux les affirmations de préférence exprimées par le groupe interrogé. Nous l'avons effectuée sur base des informations de préférences collectées lors des deuxième et troisième séances de travail. Les séances précédentes avaient permis d'obtenir une liste de huit critères d'évaluation de la priorité des substances et les performances de vingt substances sur ces huit critères, substances à utiliser comme base pour collecter les préférences. Il s'agissait de vingt substances réelles, mais dont le nom était caché aux participants.

En raison d'accords de confidentialité, nous omettons certains éléments de contexte de la présentation et nous focalisons sur la méthode suivie.

3.3.2 Mode d'interrogation

Lors de la deuxième séance, neuf participants ont donc été interrogés à partir d'une base de vingt substances. Chaque substance était décrite à l'aide d'un texte court, mais pas nommée. Le texte indiquait ensuite les performances de la substance sur chaque critère, numériquement mais également avec un commentaire textuel justifiant la note reçue par cette substance sur ce critère, ou donnant plus de détails.

Les participants devaient considérer quatre catégories de priorité possible, de la plus prioritaire (catégorie C_1) à la moins prioritaire (C_4). Chaque participant devait répartir, sans concertation avec ses voisins, pour chaque substance parmi les 20 substances proposées en exemple, sept gommettes dans les différentes catégories. Plus le nombre de gommettes est élevé dans une catégorie, plus le participant indique un degré d'accord élevé avec l'affirmation selon laquelle la substance devrait être associée à cette catégorie. Par exemple, le participant pouvait affecter la substance s_1 à la catégorie C_2 avec quatre gommettes, à la catégorie C_3 avec deux gommettes, et à la catégorie C_4 avec une gommette. L'interprétation associée serait : la substance s_1 devrait être considérée comme « assez prioritaire » (selon la sémantique associée à la catégorie C_2), avec un niveau d'accord de quatre. L'affirmation selon laquelle la substance s_1 devrait être considérée comme « faiblement prioritaire » serait associée à un niveau d'accord de deux. Enfin, l'affirmation « la substance ne me semble pas prioritaire » serait associée au plus faible niveau d'accord. L'affirmation « la substance me semble fortement prioritaire » ne serait pas

3 Convergence de modèles MR Sort pour la décision de groupe

	C_1	C_2	C_3	C_4
s_1		4	2	1
s_2	5	2		

TABLE 3.10 – Deux exemples d'évaluation de priorité de substances.

considérée comme crédible par le participant dans cet exemple. La figure 3.10 donne deux exemples d'évaluation, pour des substances s_1 et s_2 . Le participant avait pour consigne de placer au moins quatre gommettes dans une catégorie donnée, donc de choisir une catégorie majoritaire pour chaque substance. De plus, il devait fournir une répartition monotone décroissante autour de la catégorie majoritaire. Les seules combinaisons autorisées étaient donc soit de placer les sept gommettes dans une seule catégorie, soit les combinaisons $\langle 6, 1 \rangle$, $\langle 5, 2 \rangle$, $\langle 4, 3 \rangle$, ou $\langle 4, 2, 1 \rangle$, dans des catégories consécutives. Cette consigne, le respect de la monotonie, sera définie formellement à la section 4.2.2.

Ces exigences ont été expliquées à l'aide de représentations graphiques des configurations de gommettes permises et interdites, et ont été respectées par tous les participants sauf pour deux exemples.

Ce mode d'interrogation nous permettait, sans demander un effort beaucoup plus important de la part de chaque participant, d'obtenir des informations plus riches en capturant également l'hésitation possible des participants entre plusieurs catégories. Cette idée est développée plus avant au chapitre 4.

Nous avons analysé les 180 exemples fournis par les neuf participants dans le but de fournir un ou plusieurs modèles de préférence restituant leurs exemples au mieux. Cela a conduit à répartir les participants en deux groupes, selon leurs proximités de classement.

Lors de la troisième séance, chaque groupe de participants a fourni un rangement complet des vingt substances initiales. Cette fois, le rangement n'était plus individuel mais bien collectif, c'est-à-dire qu'il reflétait le consensus pour un groupe donné.

Nous ne présentons ici que l'analyse effectuée au terme de la deuxième séance. L'analyse faisant suite à la troisième séance a invoqué des modèles et outils moins directement liés aux préoccupations développées dans ce document.

3.3.3 Critères et catégories

Les participants avaient accès aux performances des substances sur les huit critères retenus lors des séances précédentes pour évaluer la priorité de traitement d'une substance. Les critères utilisent tous des échelles discrètes, souvent à trois niveaux. Certains sont binaires (oui ou non), d'autres utilisent plus que

3.3 Application réelle : échelle de risque pour produits polluants

trois niveaux.

Les participants étaient prié d'affecter chaque substance à une ou plusieurs des catégories suivantes :

C_4 La substance ne me semble pas prioritaire.

C_3 La substance me semble faiblement prioritaire.

C_2 La substance me semble assez prioritaire.

C_1 La substance me semble fortement prioritaire.

Nous n'avons pas considéré dans notre analyse suite à la deuxième séance le critère numéro huit, critère binaire. En effet, les discussions entre les participants en fin de deuxième séance semblaient indiquer que tous ne l'avaient pas interprété de la même façon. Pour certains, la valeur « oui » jouait dans le sens d'un plus fort degré de priorité, tandis que d'autres adoptaient la considération inverse. Considérer comme inconnue la direction de préférence associée par chaque participant au critère g_8 aurait nécessité pour déterminer le modèle, outre du temps, une plus grande quantité de données de préférences. En outre, rien ne garantit qu'un même participant n'aurait pas pu considérer ce critère dans un sens ou dans un autre en fonction de la substance considérée.

Les sept autres critères ont été considérés comme indiquant une priorité croissante avec des évaluations croissantes. Les critères sont donc à minimiser.

3.3.4 Choix du type de modèle

Notre analyse suite à la séance 2 s'appuie sur MR Sort sans véto. Utiliser une fonction de tri permet de « parler le même langage » que les participants : la fonction respecte l'exemple d'affectation d'un participant si elle propose d'affecter une substance à la même catégorie que celle qu'il a choisie comme la catégorie majoritaire (ce qui constitue une satisfaction idéale de cet exemple pour ce participant), ou à une catégorie parmi celles qu'il a incluses dans l'intervalle proposé à l'aide des gommettes. Le modèle MR Sort a également l'avantage de s'expliquer simplement dans les termes du participant.

3.3.5 Analyse des divergences

Une fois le modèle choisi, il convient de le paramétrer. Suivant le procédé classique de désagrégation, nous nous basons sur les données préférentielles collectées lors de la deuxième séance pour paramétrer le modèle.

Une première question concerne la prise en compte des catégories associées à un faible nombre de gommettes. Nous avons comparé deux possibilités. La

3 Convergence de modèles MR Sort pour la décision de groupe

première consiste à prendre en compte uniquement les affirmations des participants indiquées comme étant les plus crédibles. Dans ce cas, nous considérons les exemples d'affectation précis consistant à associer chaque substance à exactement une catégorie, celle où le participant a placé au moins quatre gommettes. Désignons un exemple de ce type comme associé à un niveau de crédibilité au moins 4, en référence au nombre de gommettes correspondant à la catégorie considérée. L'autre possibilité envisagée était de considérer les exemples d'affectation plus largement, donc au niveau de crédibilité 7 : un exemple d'affectation est alors l'association d'une substance à l'intervalle complet constitué des catégories où le participant a placé au moins une gomme.

L'analyse révèle un haut niveau de divergence si l'on considère uniquement les affirmations les plus crédibles, ce qui nous a conduit à choisir de considérer les intervalles d'affectation complets. Plusieurs indicateurs soutiennent ce constat.

Un premier indicateur permet de mesurer le niveau de désaccord entre participants. Il s'agit d'observer, pour chaque participant, la façon dont il ordonne les substances dans les catégories. Pour cet indicateur nous considérons l'ensemble des paires ordonnées de substances (a, b) telles que la substance a est strictement préférée à la substance b , ce qui est le cas si le participant a choisi comme catégorie majoritaire pour a une catégorie strictement meilleure que la catégorie majoritaire pour b . Ensuite, nous pouvons comparer les participants par paires. Pour chaque paire de participants, nous considérons les paires de substances (a, b) telles que a est considérée par un des participants comme meilleure que b alors que l'autre participant considère b comme meilleure que a . Cet indicateur mesure donc d'une façon minimale le désaccord, car il ne considère que les comparaisons strictes, et il ne tient pas compte d'une éventuelle perception différente de la sémantique des catégories par différents participants. La table 3.11 restitue, pour chaque paire de participants, le nombre de paires (non ordonnées) en conflit. La matrice est bien évidemment symétrique, ici ne sont indiqués que les nombres de la partie diagonale haute.

Un indicateur plus directement lié à notre objectif consiste à considérer, pour chaque paire de participants, la capacité à restituer leurs exemples, ici encore en ne considérant que la catégorie majoritaire, avec un modèle de type MR Sort sans véto. La table 3.12 indique, pour chaque paire de participants, le nombre d'erreurs minimal qu'il faut accepter pour obtenir un modèle MR Sort. Cela correspond au nombre d'exemples que le modèle ne parvient pas à restituer. Pour rappel, étant donné une paire de participants, le modèle doit restituer quarante exemples. Ce nombre d'erreurs est dit minimal car il s'agit du modèle dont les paramètres ont été choisis pour restituer au mieux les exemples fournis. Par exemple, on voit que si l'on essaye de représenter les

3.3 Application réelle : échelle de risque pour produits polluants

	Déc 2	Déc 3	Déc 4	Déc 5	Déc 6	Déc 7	Déc 8	Déc 9
Déc 1	11	22	8	16	12	18	34	10
Déc 2		21	13	27	23	11	20	11
Déc 3			34	26	28	37	50	21
Déc 4				39	4	22	10	16
Déc 5					34	22	48	28
Déc 6						30	35	20
Déc 7							13	18
Déc 8								13

TABLE 3.11 – Désaccords entre paires de participants au niveau de crédibilité 4.

	Déc 2	Déc 3	Déc 4	Déc 5	Déc 6	Déc 7	Déc 8	Déc 9
Déc 1	16	9	15	11	13	13	14	16
Déc 2		10	13	11	11	11	9	8
Déc 3			12	13	14	14	13	14
Déc 4				14	10	14	10	14
Déc 5					14	14	14	15
Déc 6						11	13	12
Déc 7							10	12
Déc 8								11

TABLE 3.12 – Nombre d’erreurs minimales au niveau de crédibilité 4.

préférences des participants numéro 1 et 2 dans un modèle commun, on ne peut que restituer, au mieux, 24 de leurs 40 exemples d’affectation, si l’on ne considère que la catégorie majoritaire qu’ils ont choisie.

L’impression fournie par cet indicateur est renforcée si l’on applique le même calcul à des sous-ensembles de participants de cardinal plus élevé que deux. Ainsi, en testant les 126 façons de choisir cinq participants parmi les neuf, et en comptant le nombre minimal d’erreurs à effectuer pour représenter le groupe de cinq participants par un même modèle de préférence, on constate que le nombre minimal d’erreurs, donc le minimum sur ces 126 possibilités, est de 36.

Ainsi, il nous a semblé plus pertinent de proposer des modèles de préférence basés sur les intervalles de catégories complets fournis par les participants, c’est-à-dire les affirmations associées à un niveau de crédibilité de 7.

La table 3.13 indique le nombre d’erreurs minimales, par paire de participants, au niveau de crédibilité 7. Il s’agit du même indicateur que celui recensé au niveau 4, à la différence que le modèle considère qu’un exemple est restitué de manière satisfaisante s’il se retrouve affecté par le modèle à une des

3 Convergence de modèles MR Sort pour la décision de groupe

	Déc 2	Déc 3	Déc 4	Déc 5	Déc 6	Déc 7	Déc 8	Déc 9
Déc 1	16	9	15	11	13	13	14	16
Déc 2		10	13	11	11	11	9	8
Déc 3			12	13	14	14	13	14
Déc 4				14	10	14	10	14
Déc 5					14	14	14	15
Déc 6						11	13	12
Déc 7							10	12
Déc 8								11

TABLE 3.13 – Nombre d’erreurs minimales au niveau de crédibilité 7.

catégories choisies par le participant, plutôt qu’à la catégorie majoritaire particulièrement.

3.3.6 Modèles proposés

L’étape suivante consiste à choisir les paramètres du modèle MR Sort permettant de reconstituer au mieux les exemples d’affectation du panel, en considérant l’entièreté des intervalles d’affectation. Si l’on cherche un modèle unique pour le groupe de neuf participants, on ne peut arriver à moins de trente erreurs, donc trente exemples d’affectation que le modèle associe à une catégorie de priorité non située parmi les catégories choisies par le participant ayant fourni cet exemple.

Nous avons donc choisi de scinder le groupe de participants en deux et de proposer deux modèles de préférence, chaque modèle associé à un sous-ensemble de participants et décrivant au mieux les préférences de ce groupe. Cela permet de réduire les erreurs (comptées alors comme la somme des erreurs effectuées par chacun des deux modèles) à dix-huit.

Nous avons donc recherché la bi-partition sur l’ensemble des participants de somme d’erreurs minimales. Quatre bi-partitions fournissent une somme d’erreurs de dix-huit. Parmi celles-ci, nous avons choisi la paire de modèles défavorisant le moins le participant le plus défavorisé, donc minimisant, sur l’ensemble des participants, le nombre d’exemples non respecté pour ce participant, ce critère étant répété lexicographiquement pour départager les solutions ex-æquo.

La bi-partition et les deux jeux de paramètres associés aux deux modèles finalement retenus, baptisés M_4 et M_5 selon le nombre de participants représentés, sont indiqués dans les tables 3.14 et 3.15. Une valeur manquante signifie que la substance n’est jamais meilleure que le profil pour cette com-

3.3 Application réelle : échelle de risque pour produits polluants

	j_1	j_2	j_3	j_4	j_5	j_6	j_7
$l^-(C_4)$	10			0	1	0	
$l^-(C_3)$	80			0	3	2	0
$l^-(C_2)$	80	1	1	2	4	3	1
w_j	0,18	0,18	0,05	0,18	0,18	0,18	0,05
$\lambda = 0,5$							

TABLE 3.14 – Modèle M_4 pour le groupe {Déc 2, Déc 3, Déc 8, Déc 9}.

	j_1	j_2	j_3	j_4	j_5	j_6	j_7
$l^-(C_4)$							
$l^-(C_3)$	90	3	2	0	4	0	0
$l^-(C_2)$	90	4	3	0	4	3	1
w_j	0,03	0,17	0,17	0,40	0,03	0,17	0,03
$\lambda = 0,5$							

TABLE 3.15 – Modèle M_5 pour le groupe {Déc 1, Déc 4, Déc 5, Déc 6, Déc 7}.

paraison. Ces deux modèles parviennent à restituer 162 exemples d'affectation sur les 180 donnés. Rappelons que les critères sont à minimiser. Donc une substance parvient au niveau de priorité correspondant à la catégorie C si la substance a des valeurs de performances égales ou inférieures au profil $l^-(C)$ correspondant, sur un nombre de critères tel que la somme des poids parvient à $\lambda = 0,5$. Par exemple, pour le modèle KP, YG, CS, MC, une substance qui aurait des performances $\langle 80, 1, 2, 2, 2, 0, 0 \rangle$ serait classée dans la catégorie C_3 : elle est au moins aussi bonne que le profil $l^-(C_2)$ (c'est-à-dire a des valeurs de performances égales ou inférieures) sur les critères $j_1, j_2, j_3, j_5, j_6, j_7$ (somme des poids = $0,82 \geq 0,5$) et est au moins aussi bonne que le profil $l^-(C_3)$ sur les critères j_1, j_5, j_6, j_7 (somme des poids = $0,59 \geq 0,5$), mais elle ne peut atteindre la catégorie C_4 car seul le critère j_6 y est favorable (somme des poids = $0,18 < 0,5$).

3.3.7 Affectations par ces modèles

Les affectations que chacun de ces modèles entraîne, pour chaque substance, sont indiquées en Table 3.16.

3.3.8 Discussion

Cette analyse a permis de déterminer, en utilisant une variante du programme ICL, deux ensembles de participants aux raisonnements apparemment

3 Convergence de modèles MR Sort pour la décision de groupe

	s_1	s_2	s_3	s_4	s_5	s_6	s_7	s_8	s_9	s_{10}
M_4	2	2	1	2	1	1	2	1	1	3
M_5	3	2	2	3	2	2	3	2	3	2
	s_{11}	s_{12}	s_{13}	s_{14}	s_{15}	s_{16}	s_{17}	s_{18}	s_{19}	s_{20}
M_4	1	4	4	3	2	3	1	2	2	3
M_5	2	3	3	2	2	2	2	2	2	3

TABLE 3.16 – Les affectations résultantes de l’application des modèles inférés.

proches. Lors de la séance trois, chaque sous-groupe a travaillé à l’élaboration d’un rangement complet des vingt substances. Ces deux rangements ont à leur tour servi de base à l’élaboration de modèles de rangement.

Notre objectif en choisissant de regrouper les participants ayant des raisonnements similaires pour la séance trois était d’augmenter les chances que le rangement fourni reflète la façon de penser de chaque membre du groupe. Si au contraire un groupe de gens est composé d’individus aux raisonnements très hétérogènes, le rangement résultant de la discussion de groupe contiendra beaucoup de compromis difficiles et risque de représenter une moyenne de compromis en des sens divers. Un tel agrégat de compromis ne représente l’avis de personne et aurait donc fourni peu d’informations exploitables.

Ici encore se pose une question de validation difficile. Des tests répétés devraient être effectués pour mesurer si le rangement obtenu à l’issue du processus reflète mieux l’avis de chaque membre de chaque sous-groupe, selon la façon dont on a déterminé les sous-groupes au préalable. Ici, les sous-groupes sont déterminés en fonction de la somme des exemples d’affectation non reproductibles dans un modèle MR Sort sans véto. Il serait intéressant de comparer cette façon de déterminer les sous-groupes à, par exemple, des procédés visant à déterminer les sous-groupes en s’appuyant sur des modèles différents.

4 Crédibilités valuées et exploitation simple

Ce chapitre présente une procédure de collecte et d'exploitation d'informations applicable à une situation mono-décideur.

4.1 Contexte

Supposons qu'un décideur faisant face à un problème de tri souhaite utiliser la classe de modèles de tri M pour guider ses choix d'affectation. Une approche classique pour aider un décideur est de capturer ses préférences de manière robuste, en considérant l'ensemble des modèles de préférence compatibles avec ses affirmations de préférence.

Notons ω un ensemble de valeurs de paramètres pour la classe de modèles de tri M , déterminant donc une fonction de tri $f_\omega \in M$. Dans cette section, la classe de modèles choisie peut être précise ou imprécise, c'est-à-dire que la fonction de tri peut être précise ($f_\omega : \mathbb{A} \mapsto \mathcal{C}$) ou imprécise ($f_\omega : \mathbb{A} \mapsto 2^{\mathcal{C}}$). Considérons qu'elle est imprécise. Nous pouvons le faire sans perte de généralité, car une fonction de tri précise peut être redéfinie comme une fonction imprécise n'utilisant que des singletons.

Notons Ω^* l'ensemble des paramétrages possibles pour M , donc Ω^* est un ensemble de modèles de préférence ω . Le décideur peut contraindre l'ensemble de modèles candidats à la représentation de ses préférences en donnant des informations telles que des exemples d'affectation. Notons Ω l'ensemble de modèles de préférence satisfaisant les contraintes fournies, $\Omega \subseteq \Omega^*$. Ceci définit une fonction de tri robuste $f_\Omega : \mathbb{A} \mapsto 2^{\mathcal{C}}$, telle que $\forall a \in \mathbb{A}$:

$$f_\Omega(a) = \bigcup_{\omega \in \Omega} f_\omega(a).$$

Le décideur peut souhaiter donner uniquement des informations dont il est certain, telles que des exemples d'affectation imprécis, associant à une alternative un intervalle de catégories possibles, ou des contraintes peu précises sur les poids, par exemple en indiquant que tel poids devrait être plus important que tel autre poids. Cela conduit à considérer un espace de modèles compatibles

4 Crédibilités évaluées et exploitation simple

Ω très vaste, qui conduira à une fonction de tri robuste très peu précise : la fonction de tri risque d'affecter chaque alternative à un ensemble de catégories possibles très important et les conclusions seront donc pauvres.

À l'autre extrême, le décideur peut souhaiter contraindre fortement l'espace des modèles compatibles, de manière à obtenir des recommandations précises. Cela risque cependant d'exiger un travail important et difficile pour le décideur, car déterminer exactement ses propres préférences n'est pas chose facile. Une telle approche comporte également un risque de diminution de la qualité des informations données par le décideur. Si l'analyste pose des questions exigeantes, le décideur peut être tenté de répondre alors même qu'il n'est pas sûr de sa réponse. Enfin, les décideurs sont des gens très occupés. La presse du jour, au moment où ces lignes sont écrites, fait part de la volonté de citoyens d'inciter les responsables politiques à refuser le cumul des mandats en France. L'Internationale de l'AMCD devrait peut-être appuyer ce mouvement de manière à augmenter la disponibilité des décideurs. Malheureusement, même quand il n'a « qu'un » mandat, il est souvent impossible de garder un décideur sous la main suffisamment longtemps pour obtenir des réponses précises à de nombreuses questions.

Nous proposons dans ce chapitre une méthode permettant d'obtenir des informations de préférences riches de la part du décideur, sans lui poser pour autant des questions difficiles. Ces informations nous permettent ensuite d'obtenir plusieurs fonctions de tri robustes, associées à différents degrés de crédibilité, et fournissant des recommandations plus ou moins précises en fonction du degré de crédibilité souhaité.

Nous demandons au décideur d'associer à chaque alternative prise comme exemple et chaque catégorie possible une crédibilité indiquant si l'alternative peut être affectée à cette catégorie. Cela permet de définir, pour un niveau de crédibilité donné, un ensemble d'exemples d'affectation plus ou moins précis, de telle manière que le niveau de crédibilité le plus élevé contienne les exemples les moins précis. Ainsi, le niveau de crédibilité le plus élevé détermine un ensemble de modèles de préférence compatibles $\Omega^{\text{sûr}}$ très vaste, associé à une fonction de tri robuste $f_{\Omega^{\text{sûr}}}$ peu précise. Au contraire, le niveau de crédibilité le plus faible est celui qui détermine l'ensemble de modèles de préférence compatibles Ω^{douteux} le plus petit, et conduit donc à une fonction de tri robuste associée $f_{\Omega^{\text{douteux}}}$ plus précise. Quand une nouvelle alternative a doit être triée, le décideur peut alors prendre connaissance d'indications de différentes crédibilités. Les affectations fournies par $f_{\Omega^{\text{sûr}}}$ sont associées à un niveau de crédibilité élevé, mais peu précises, alors que celles données par $f_{\Omega^{\text{douteux}}}$ sont moins crédibles mais plus précises.

La section suivante décrit la procédure proposée pour collecter les exemples

d'affectation associés à des crédibilités. Cela constitue l'entrée de la procédure d'exploitation, décrite à la section 4.3. Un exemple illustratif est proposé en Section 4.4. Les procédures que nous proposons sont indépendantes de la classe de modèles de tri choisie M , mais nous illustrerons l'approche en utilisant le modèle MR Sort.

4.2 La procédure Argument Strength Assessment (ASA)

4.2.1 ASA ordinale

La procédure suivante, ASA, vise à obtenir du décideur des exemples d'affectation valués, c'est-à-dire des exemples où une affectation possible d'une alternative à une catégorie est associée à un degré de confiance indiquant la crédibilité que l'alternative soit affectée à cette catégorie.

Les exemples d'affectation valués utilisés dans la suite de cette partie sont définis précisément comme suit.

Définition 1. *Considérons un ensemble d'alternatives X utilisées comme exemples et un ensemble de catégories \mathcal{C} . Un **exemple d'affectation valué** v_x de $x \in X$ est une application de \mathcal{C} vers un ensemble T de niveaux de crédibilité muni d'un ordre total $>_T$. L'ensemble T est borné, donc admet un niveau de crédibilité minimum noté 0 et un niveau maximum noté t . La valeur $v_x(C)$ indique la crédibilité de l'affectation de x à C . Une crédibilité $v_x(C) = 0$ indique que l'alternative x ne doit certainement pas être affectée à C , et une crédibilité $v_x(C) = t$ indique que l'alternative x est certainement affectée à C .*

Dans la suite de l'exposé, nous considérerons par simplicité que T est l'ensemble des entiers naturels borné par une valeur $t \in \mathbb{N}$ fixée a priori, et utiliserons l'ordre habituel $>$ sur les entiers au lieu de $>_T$. L'approche que nous proposons s'applique également si l'on considère par exemple un ensemble d'étiquettes {« douteux », « crédible », « sûr »}. Nous présentons également, dans la section 4.2.3, une manière alternative de définir ASA, de manière cardinale.

Nous utiliserons également la notation v pour dénoter une application de $X \times \mathcal{C}$ vers \mathbb{N} définie comme $v(x, C) = v_x(C)$. L'application v représente l'ensemble de valeurs de crédibilités fourni par le décideur, et servira d'entrée à la procédure d'exploitation.

- Le décideur est d'accord avec les interprétations suivantes de ces crédibilités.
- $v_x(C) > v_x(C')$ ssi les arguments en faveur de l'affectation de x à C sont plus forts que les arguments en faveur d'une affectation de x à C' . Cette dernière affirmation est considérée comme la moins crédible des deux.

4 Crédibilités valuées et exploitation simple

	C_1	C_2	C_3	C_4
x_1	0	2	4	3
x_2	6	2	1	0

TABLE 4.1 – Deux exemples d'affectation valués.

- $v_x(C) = v_x(C')$ ssi les arguments en faveur de l'affectation de x à C sont équivalents à la force des arguments en faveur de l'affectation de x à C' . Ces deux affirmations sont considérées comme également crédibles.
- $v_x(C) = 0$ ssi le décideur considère l'affectation de x à C inacceptable. Le décideur est certain que C n'est pas une catégorie adéquate pour x .
- $v_x(C) = t$ ssi le décideur considère impossible l'affectation de x à C' pour tout $C' \neq C$. Le décideur est sûr que C est la seule catégorie adéquate pour x .
- $v_x(C) > v_y(C')$ ssi les arguments en faveur de l'affectation de x à C sont plus forts que les arguments en faveur de l'affectation de y à C' . La première affirmation est considérée plus crédible.

Exemple 1. *Considérons les crédibilités données par le décideur et reproduites en Table 4.1. Ces exemples indiquent que le décideur ne considère pas l'affectation de x_1 en C_1 crédible. L'affectation la plus crédible pour x_1 est C_3 , tandis que $x_1 \rightarrow C_4$ serait le deuxième choix et $x_1 \rightarrow C_2$ la possibilité la moins crédible des trois. De la même manière, la catégorie C_4 est la destination la moins crédible pour x_2 et C_1 est la plus crédible.*

4.2.2 Propriétés ordinales

Soit $x \in X$ une alternative utilisée comme exemple. Les propriétés suivantes peuvent être satisfaites par un exemple d'affectation valué v_x .

Définition 2. v_x est **net** s'il existe $C \in \mathcal{C}$ tel que

$$v_x(C) = t.$$

Un exemple d'affectation valué net représente une affectation d'une alternative dans exactement une catégorie. La figure 4.1 illustre cette situation.

Définition 3. v_x est **unimodal** s'il existe $C \in \mathcal{C}$ telle que $\forall C_i, C_j \in \mathcal{C}$:

$$\begin{cases} v_x(C_i) \leq v_x(C_j) < v_x(C) \text{ si } C_i \triangleleft C_j \triangleleft C \\ \text{et} \\ v_x(C_i) \leq v_x(C_j) < v_x(C) \text{ si } C \triangleleft C_j \triangleleft C_i \end{cases}$$

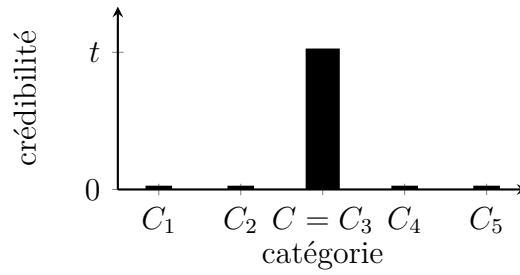


FIGURE 4.1 – Un exemple d'affectation valué net.

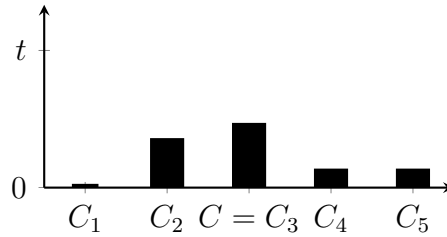


FIGURE 4.2 – Un exemple d'affectation valué unimodal.

Un exemple d'affectation valué unimodal (représenté en Figure 4.2) désigne une unique catégorie telle que les crédibilités des autres catégories ne décroissent pas lorsqu'on s'approche de cette catégorie.

Définition 4. v_x est **faiblement unimodal** (représenté en Figure 4.3) s'il existe $C \in \mathcal{C}$ telle que $\forall C_i, C_j \in \mathcal{C}$

$$\begin{cases} v_x(C_i) \leq v_x(C_j) \leq v_x(C) \text{ si } C_i \triangleleft C_j \triangleleft C \\ \text{et} \\ v_x(C_i) \leq v_x(C_j) \leq v_x(C) \text{ si } C \triangleleft C_j \triangleleft C_i \end{cases}$$

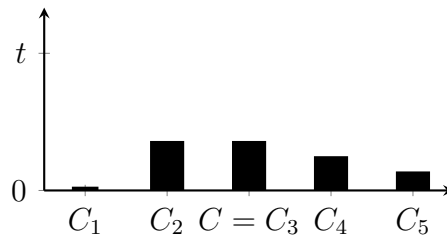


FIGURE 4.3 – Un exemple d'affectation valué faiblement unimodal.

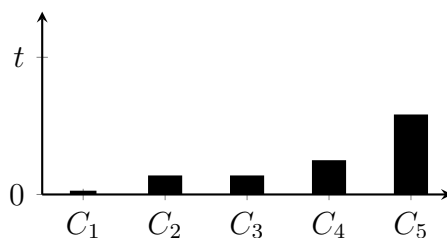


FIGURE 4.4 – Un exemple d’affectation valué monotone.

Définition 5. v_x est *monotone* si

$$\left\{ \begin{array}{l} v_x(C_i) \leq v_x(C_j) \quad \forall C_i \triangleleft C_j \\ \text{ou} \\ v_x(C_j) \leq v_x(C_i) \quad \forall C_i \triangleleft C_j. \end{array} \right.$$

Dans ce cas, les crédibilités sont non croissantes ou non décroissantes depuis la pire catégorie vers la meilleure catégorie. Voir Figure 4.4. Lorsqu’un exemple est monotone, il est faiblement unimodal.

4.2.3 ASA cardinale

Dans la variante cardinale de ASA, on désire s’assurer que la somme des crédibilités pour chaque exemple est constante. Cette variante ne peut être définie lorsque T représente un ensemble d’étiquettes, mais s’applique par exemple lorsque l’ensemble de niveaux de crédibilité est un ensemble d’entiers.

Pour chaque alternative x , le décideur alloue un nombre de points fixé t parmi les différentes catégories de \mathcal{C} . Le nombre de points associé à une catégorie $C \in \mathcal{C}$ représente la crédibilité que le décideur associe à l’affectation de x à C .

Par exemple, avec $t = 10$, si le décideur est certain que l’alternative x_1 doit être associée à une catégorie donnée C_h , il alloue les dix points à C_h et zéro points aux autres catégories. S’il considère comme également crédible que x_2 soit affectée à C_i ou à C_{i+1} , mais aucunement possible que x_2 aille dans une autre catégorie que celles-là, il allouera cinq points à C_i et cinq points à C_{i+1} . Il ne peut pas, dans cette version, attribuer trois points à C_i et trois points à C_{i+1} , car la somme n’atteindrait pas t .

Lorsque les niveaux de crédibilité sont entiers, nous pouvons définir un **exemple d’affectation valué additif** v_x^{additif} en utilisant l’additivité des entiers.

Définition 6. *Considérons un ensemble d’alternatives X utilisées comme exemples et un ensemble de catégories \mathcal{C} . Un **exemple d’affectation va-***

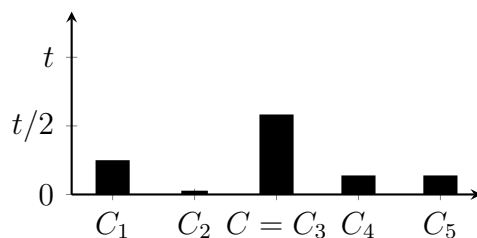


FIGURE 4.5 – Un exemple d’affectation valuée admettant une majorité.

lué additif v_x^{additif} de $x \in X$ est une application de \mathcal{C} vers \mathbb{N} telle que

$$\begin{cases} v_x^{\text{additif}}(C) \in \{0, \dots, t\}, \quad \forall C \in \mathcal{C} \\ \sum_{C \in \mathcal{C}} v_x^{\text{additif}}(C) = t. \end{cases}$$

Nous pouvons alors définir une nouvelle propriété.

Définition 7. v_x **admet une majorité** s’il existe $C \in \mathcal{C}$ telle que

$$v_x(C) > t/2$$

Dans ce cas, représenté en Figure 4.5, le décideur considère que les arguments en faveur de l’affectation de x en catégorie C sont plus forts que les arguments conduisant à affecter x à n’importe quelle autre catégorie.

La suite de ce chapitre s’applique indépendamment que l’on considère des exemples d’affectation valués ou des exemples d’affectation valués additifs.

4.3 Exploitation

Cette section présente une procédure d’exploitation simple d’exemples d’affectation valués.

Les sections 4.3.1 et 4.3.2 décrivent le fonctionnement de la procédure sur un exemple simple, en considérant l’exploitation de l’exemple donné ci-dessus (Table 4.1). La procédure est ensuite décrite formellement à la section 4.3.3.

4.3.1 Illustration de l’exploitation : ensemble X

La première partie de la procédure d’exploitation transforme v en plusieurs ensembles d’exemples d’affectation. L’ensemble d’exemples le moins contraignant est $\{x_1 \rightarrow [C_2, C_4]; x_2 \rightarrow [C_1, C_3]\}$; l’ensemble d’exemples le

4 Crédibilités évaluées et exploitation simple

plus contraignant, mais le moins crédible, est $\{x_1 \rightarrow [C_3]; x_2 \rightarrow [C_2]\}$. La notation Γ_X désigne un tel ensemble d'exemples d'affectation, par exemple $\Gamma_X = \{x_1 \rightarrow [C_2, C_4]; x_2 \rightarrow [C_1, C_3]\}$. Chacun de ces ensembles est associé à un niveau de crédibilité.

Les paires d'ensemble d'exemples et de niveau de crédibilité construites par la première partie de la procédure d'exploitation sont indiquées en Figure 4.6. La figure montre un graphe dont les nœuds correspondent chacun à un niveau de crédibilité associé à un ensemble d'exemples d'affectation. La procédure commence par exclure des affectations possibles les catégories associées au niveau de crédibilité le plus bas, à savoir zéro. Ceci donne des affectations de x_1 à $[C_2, C_4]$ et de x_2 à $[C_1, C_3]$, indiquées dans le nœud au sommet du graphe. Le niveau de crédibilité associé est le complément à t du niveau de crédibilité exclu, à savoir à cette étape, le complément de zéro à neuf. La procédure exclut ensuite le niveau de crédibilité suivant, ici le niveau un, et exclut donc l'affectation possible de x_2 à C_3 . À l'itération suivante, les affectations de crédibilité inférieure ou égale à deux sont exclues.

La construction des nœuds s'achève avec l'exclusion des crédibilités inférieures ou égales à trois. En effet, exclure le niveau quatre est impossible car cela conduirait à refuser d'affecter x_1 à une quelconque catégorie. Les arcs du graphe de la figure 4.6 relient les nœuds de crédibilité forte aux nœuds de crédibilité plus faible, dans l'ordre utilisé par la procédure. Les arcs déductibles par transitivité ne sont pas indiqués.

Nous utilisons ici la notion de complément à t pour associer des niveaux de crédibilité aux différents ensembles d'exemples déduits de v . Lorsque l'ensemble des niveaux de crédibilité T n'est pas inclus dans \mathbb{N} , cette opération n'est pas nécessairement définie. Cependant, cette opération n'est utilisée ici que comme un moyen commode d'inverser l'ordre des niveaux de crédibilité $>_T$, de manière à associer la crédibilité la plus élevée à l'ensemble d'exemples résultant de l'exclusion des exemples les moins crédibles. L'opération de calcul du complément peut être remplacée par n'importe quelle numérotation des ensembles d'exemples résultants indiquant que la crédibilité des ensembles successifs est décroissante.

4.3.2 Illustration de l'exploitation : ensemble A

Cette section montre le fonctionnement de la deuxième partie de la procédure d'exploitation sur l'exemple. Il s'agit maintenant d'exploiter les ensembles d'exemples obtenus lors de la première partie pour trier un autre ensemble d'alternatives.

Définissons un nouvel ensemble d'alternatives A contenant des alternatives a_1, \dots, a_5 que le décideur souhaite trier. Afin de définir ses préférences, le

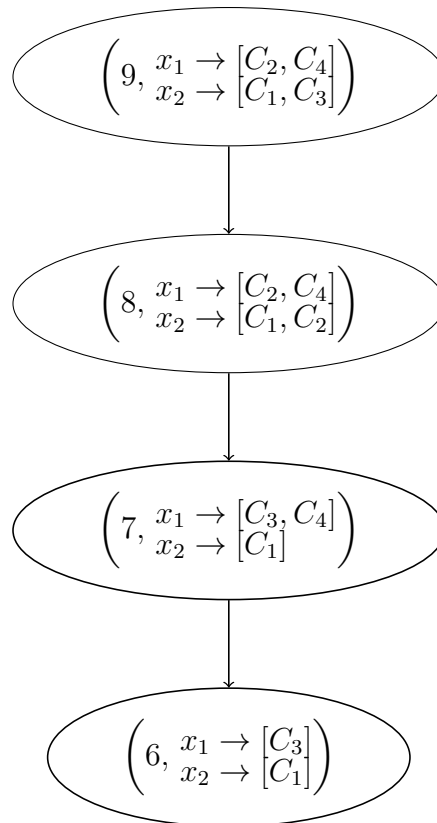


FIGURE 4.6 – Les ensembles d'exemples d'affectation extraits des crédibilités fournies (voir Table 4.1).

4 Crédibilités évaluées et exploitation simple

	$g1$	$g2$	$g3$
x_1	46	12	48
x_2	32	87	96
a_1	57	42	84
a_2	17	85	43
a_3	26	32	92
a_4	41	2	28
a_5	53	84	77

TABLE 4.2 – Les performances des alternatives exemples et des alternatives à trier.

décideur a fourni les deux exemples indiqués ci-avant x_1 et x_2 . Les alternatives sont évaluées sur trois critères et leurs performances sont reprises dans la table 4.2. Afin de trier les alternatives, le décideur souhaite utiliser la classe de modèles $M = \text{MR Sort sans véto}$. Notons que la procédure que nous proposons ne requiert pas l'utilisation d'une classe de modèle en particulier.

Considérons les ensembles d'exemples d'affectation indiqués à la figure 4.6. Chaque ensemble d'exemples Γ peut être utilisé pour définir des contraintes sur l'ensemble Ω^* des modèles de préférence possibles, et donc, pour définir les affectations possibles pour une alternative $a \in A$. Considérons, par exemple, $\Gamma = \{x_1 \rightarrow [C_2, C_4]; x_2 \rightarrow [C_1, C_3]\}$, et observons si l'alternative a_1 peut être affectée à la catégorie C_1 . Ceci revient à demander s'il existe un modèle de préférence $\omega \in \Omega^*$ satisfaisant les deux exemples x_1 et x_2 et affectant a_1 à C_1 . Les PMs vus en section 3.1 permettent de répondre à cette question. Ici, la réponse est négative : il est impossible de trouver un modèle de préférence qui affecte a_1 à C_1 et satisfait les exemples. Voyons maintenant si a_1 peut être affectée à C_2 . Le PM peut trouver un modèle de préférence satisfaisant dans ce cas : utiliser un modèle ω avec des profils $l(C_2) = \langle 26, 0, 41 \rangle$; $l(C_3) = \langle 62, 48, 82 \rangle$; $l(C_4) = \langle 64, 105, 103 \rangle$ et des poids $\langle 0, 1, 0 \rangle$ entraîne $f_\omega(x_1) = C_2$, $f_\omega(x_2) = C_3$, $f_\omega(a_1) = C_2$. On en conclut que $a_1 \rightarrow C_2$ est compatible avec les exemples d'affectation Γ . De même, on peut observer que a_1 peut être affectée en C_3 . Ce raisonnement s'applique à toutes les alternatives $a \in A$ et catégories $C \in \mathcal{C}$, et conduit, pour chaque alternative, à trouver les affectations compatibles avec $\Gamma = \{x_1 \rightarrow [C_2, C_4]; x_2 \rightarrow [C_1, C_3]\}$. Ils sont donnés par la table 4.3. Ces conclusions peuvent être associées à la même crédibilité que les exemples d'affectation Γ , à savoir 9, car seules des informations de ce niveau de crédibilité ont été utilisées pour les obtenir.

L'algorithme peut être répété pour chaque Γ de la figure 4.6. Les résultats associés au niveau de crédibilité 8 sont donnés par la table 4.4. Considérons

	C_1	C_2	C_3	C_4
a_1		✓	✓	✓
a_2	✓	✓	✓	
a_3	✓	✓	✓	
a_4	✓	✓	✓	✓
a_5		✓	✓	✓

TABLE 4.3 – Affectations possibles au niveau de crédibilité 9.

	C_1	C_2	C_3	C_4
a_1		✓	✓	✓
a_2	✓	✓		
a_3	✓	✓		
a_4	✓	✓	✓	✓
a_5		✓	✓	✓

TABLE 4.4 – Affectations possibles au niveau de crédibilité 8.

l'alternative a_2 : au niveau de crédibilité 9, elle peut être affectée à C_1 , C_2 ou C_3 , tandis qu'au niveau de crédibilité 8, elle ne peut plus être affectée qu'à C_1 ou C_2 . Cela illustre que les affectations peuvent devenir plus précises lorsque le niveau de crédibilité exigé baisse. Tous les résultats sont affichés en Figure 4.7.

Le décideur peut alors choisir d'utiliser les résultats associés à une très grande crédibilité, ou d'utiliser des résultats plus précis, bien que moins crédibles. Notons que le graphe résultant présente la propriété intéressante que, pour toute paire de nœuds n_1, n_2 , si n_2 est moins crédible, alors il est aussi plus précis, dans le sens où les ensembles de catégories où chaque alternative peut être affectée sont des sous-ensembles des ensembles correspondants dans n_1 . Cette propriété permet au décideur de visualiser et interpréter les résultats facilement. Nous verrons en section 4.4 une façon possible de montrer au décideur les recommandations obtenues.

4.3.3 Description formelle

Cette section présente une description formelle de la procédure.

Partie 1 : construction des fonctions de tri robustes sur X

Définissons \bar{e} comme le minimum, sur tous les exemples $x \in X$, de la crédibilité maximale sur chaque catégorie : $\bar{e} = \min_{x \in X} \max_{C \in \mathcal{C}} v_x(C)$. L'ensemble B correspond aux valeurs de crédibilité qui doivent être considérées :

4 Crédibilités valuées et exploitation simple

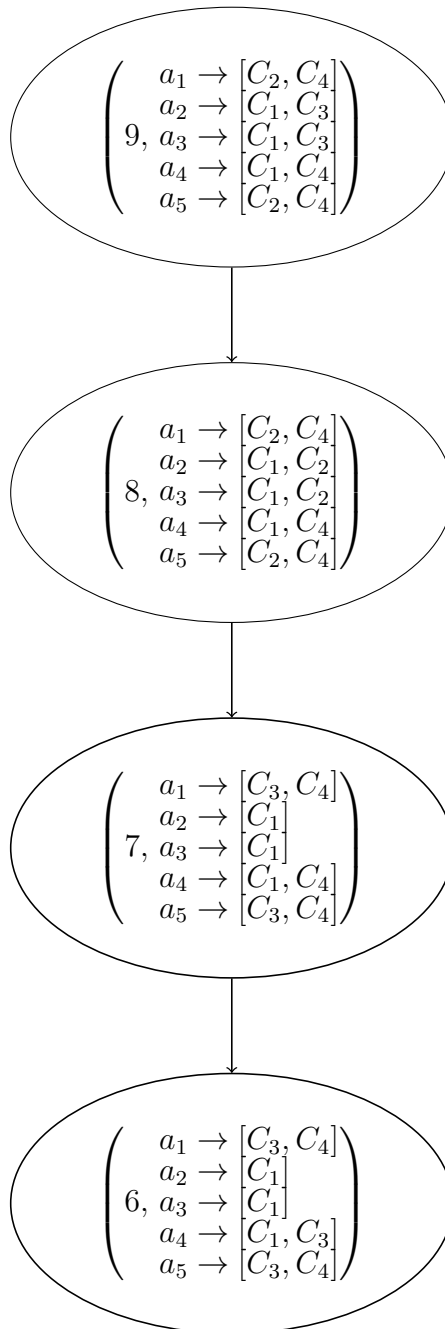


FIGURE 4.7 – Affectations résultantes de l'exploitation.

$B = \{b \in \mathbb{N} \mid b = t - e, \forall e \in 0, 1, \dots, \bar{e}\}$. La procédure exclura les niveaux de crédibilité $0, 1, \dots, \bar{e}$. Définissons, $\forall b \in B$, une fonction de tri robuste $\Gamma_X^b : X \mapsto 2^{\mathcal{C}}$ sur l'ensemble d'exemples X comme suit :

$$\Gamma_X^b(x) = \{C \in \mathcal{C} \mid v_x(C) > t - b\}.$$

La première partie de la procédure calcule donc \bar{e} et construit toutes les fonctions Γ_X^b .

Partie 2 : construction des fonctions de tri robustes sur A

La deuxième partie requiert d'avoir choisi une classe de modèles M et un ensemble d'alternatives à trier A . L'ensemble des modèles de préférence possibles pour M est noté Ω^* . La partie 2 utilise les fonctions de tri robustes Γ_X^b définies en partie 1 pour construire des fonctions d'affectation robustes Γ_A^b sur A , pour chaque niveau de crédibilité b .

$$\Omega^{(\Gamma_X^b)} = \{\omega \in \Omega^* \mid \forall x \in X : f_\omega(x) \in \Gamma_X^b(x)\}.$$

L'ensemble $\Omega^{(\Gamma_X^b)}$ représente l'ensemble des modèles de préférence ω satisfaisant les exemples d'affectation définis par Γ_X^b .

Si $\Omega^{(\Gamma_X^b)} = \emptyset$, alors il n'y a pas de modèle de préférence $\omega \in \Omega^*$ satisfaisant les exemples Γ_X^b . Le reste de la procédure ne considère pas de tels ensembles d'exemples Γ_X^b .

Pour chaque Γ_X^b tel que $\Omega^{(\Gamma_X^b)} \neq \emptyset$, définissons une fonction d'affectation robuste $\Gamma_A^b : A \mapsto 2^{\mathcal{C}}$ comme suit :

$$\Gamma_A^b(a) = \bigcup_{\omega \in \Omega^{(\Gamma_X^b)}} f_\omega(a).$$

$\Gamma_A^b(a)$ représente l'ensemble des catégories où l'alternative a peut être affectée, en utilisant des modèles de préférence satisfaisant les exemples d'affectation définis par Γ_X^b . Notons Γ_A l'ensemble de paires constituées d'un niveau de crédibilité et d'une fonction d'affectation robuste :

$$\Gamma_A = \{(b, \Gamma_A^b), \forall b \in B\}.$$

Partie 3 : élagage

Γ_A peut contenir des fonctions d'affectation robustes identiques $\Gamma_A^b = \Gamma_A^{b'}$ pour différents niveaux de crédibilité b, b' . Dans un tel cas, seule la fonction d'affectation correspondant à la crédibilité la plus élevée devrait être montrée au décideur, l'autre n'apportant pas d'informations supplémentaires. Nous obtenons dès lors l'ensemble Γ'_A comme suit :

$$\Gamma'_A = \{(b, \Gamma_A^b) \in \Gamma_A \mid \nexists (b', \Gamma_A^{b'}) \in \Gamma_A \text{ tel que } \Gamma_A^b = \Gamma_A^{b'} \wedge b < b'\}.$$

4.3.4 Propriété de fonctions d'affectations emboîtées

Une propriété intéressante de la procédure est qu'elle affecte les alternatives à des ensembles de catégories emboîtés, si on les considère ordonnés par le niveau de crédibilité associé.

Proposition 1. $\forall (b, \Gamma_A^b), (b', \Gamma_A^{b'}) \in \Gamma'_A :$

$$b \leq b' \Leftrightarrow \forall a \in A : \Gamma_A^b(a) \subseteq \Gamma_A^{b'}(a).$$

La preuve découle immédiatement de la construction de la procédure.

Cela signifie que lorsque le décideur regarde des résultats moins crédibles, il obtient, en contrepartie, des recommandations plus précises.

4.3.5 Performance des calculs avec MR Sort

L'exploitation des crédibilités requiert de vérifier, pour chaque niveau de crédibilité et pour chaque alternative dans A et catégorie dans \mathcal{C} , si l'alternative peut être affectée à cette catégorie étant donné les exemples d'affectation au niveau de crédibilité choisi. Lorsque le modèle utilisé est MR Sort, chacun de ces tests requiert la résolution d'un PLNE. Nous avons effectué des tests de performance afin d'évaluer quelle taille de problèmes peut être résolu en un temps raisonnable. La méthode de test est indiquée à la section 3.1.7. La table 4.5 montre le temps de résolution moyen d'un test, donc le temps pour tester s'il est possible d'affecter une alternative donnée de A à une catégorie donnée de \mathcal{C} , étant donné un niveau de crédibilité déterminé. Le temps est donné en millisecondes. Par exemple, si vingt alternatives doivent être triées dans trois catégories sur base de dix exemples d'affectation, en prenant en compte cinq critères, et en considérant trois niveaux de crédibilité, le nombre de tests requis serait $20 \times 3 \times 3 = 180$ et temps approximatif de résolution de tous les PMs serait $180 \text{ tests} \times 193 \text{ millisecondes par test} = 35 \text{ secondes}$. Les temps affichés représentent uniquement le temps de résolution du PM par le solveur et ne tiennent pas compte du temps de construction de l'ensemble des variables et contraintes, qui pourrait ne pas être négligeable dans le cas de nombreux très petits problèmes à résoudre. Il faut aussi noter que l'implémentation actuelle n'est pas optimisée pour ce genre d'usage, une nette amélioration est donc probablement possible.

Les résultats affichés montrent que, lorsque le modèle choisi est MR Sort, de petits problèmes peuvent être résolus avec les PMs existants, mais l'approche ne s'étend pas à des problèmes plus gros. Ceci est particulièrement vrai si la procédure doit être utilisée de manière interactive.

	10	20	30
3	65	179	219
4	77	356	753
5	193	672	2604
6	729	3942	19 528
7	3428	8371	152 397

TABLE 4.5 – Temps moyen pour effectuer un test d’affectation, en millisecondes, en fonction du nombre d’exemples (en colonne) et du nombre de critères (en ligne).

Le problème peut être surmonté de plusieurs manières. Comme indiqué précédemment, des méthodes de résolutions prenant en compte la structure particulière du problème à résoudre s’avèreraient probablement beaucoup plus rapides. Réduire le nombre de variables dans les PMs réduit également le temps de résolution, en particulier, si les performances des profils ou des poids sont déterminées au préalable, les problèmes à résoudre deviennent purement linéaires, ce qui réduit considérablement le temps de résolution.

Par ailleurs, l’utilisation de modèles de tri autres que MR Sort conduirait à d’autres temps de résolution. En particulier, utiliser des fonctions de valeur au lieu d’MR Sort conduirait à résoudre des PMs purement linéaires (Greco *et al.*, 2008).

4.4 Exemple illustratif

Illustrons la procédure proposée sur un exemple illustratif. Supposons que la responsable d’une faculté doive évaluer cent étudiants de sa faculté à la fin de l’année, en fonction de leurs notes dans trois cours : mathématique (g_1), physique (g_2) et littérature (g_3). Pour chaque cours, la note est un entier entre un, la moins bonne note, et quatre, la meilleure note. Pour chaque étudiant, la responsable doit choisir une possibilité parmi quatre : l’étudiant échoue (C_1), il doit passer des examens supplémentaires (C_2), il réussit (C_3), ou il réussit avec mention (C_4). La responsable souhaite utiliser le modèle MR Sort pour guider le processus d’évaluation.

La responsable commence par fournir dix exemples valués d’évaluation d’étudiants qu’elle connaît. Les notes de ces étudiants pour les trois cours sont données en Table 4.6 et les exemples valués sont indiqués dans la table 4.7. La responsable répartit, pour chaque étudiant, sept points de crédibilité entre les différentes catégories où il peut être associé. Par exemple, la responsable considère que l’étudiant x_{01} , noté $\langle 1, 4, 3 \rangle$ pour les cours de mathématique, physique

4 Crédibilités valuées et exploitation simple

	g_1	g_2	g_3
x_{01}	1	4	3
x_{02}	1	1	4
x_{03}	3	1	4
x_{04}	2	1	4
x_{05}	2	2	2
x_{06}	3	1	2
x_{07}	4	3	2
x_{08}	1	3	3
x_{09}	4	3	1
x_{10}	1	4	4

TABLE 4.6 – Évaluations des exemples d'affectation.

	C_1	C_2	C_3	C_4
x_{01}	0	2	4	1
x_{02}	7	0	0	0
x_{03}	0	2	4	1
x_{04}	1	4	2	0
x_{05}	0	7	0	0
x_{06}	0	7	0	0
x_{07}	0	0	3	4
x_{08}	3	4	0	0
x_{09}	0	2	4	1
x_{10}	0	1	4	2

TABLE 4.7 – Exemples d'affectation valués initiaux.

et littérature, devrait, le plus vraisemblablement, réussir (catégorie C_3), mais il serait acceptable qu'il soit affecté en catégorie C_2 , donc qu'il doive passer des examens supplémentaires, ou, avec la crédibilité la plus basse, il pourrait se voir attribuer une mention, donc être affecté en catégorie C_4 . L'étudiant ne devrait certainement pas échouer, comme l'indique la valeur de crédibilité nulle associée à la catégorie C_1 .

La procédure peut être appliquée pour calculer les affectations de l'ensemble des cent étudiants à trier, en considérant les exemples d'affectation valués au niveau de crédibilité sept, puis six, c'est-à-dire, en excluant les affectations possibles de niveau de crédibilité zéro, puis un. Les affectations des dix premiers étudiants, notés a_{001} à a_{010} , ainsi que les exemples, sont affichés en Figure 4.8. Les données correspondant à cet exemple sont disponibles en format électronique (voir annexe D, numéro 3).

Les exemples associés au niveau de crédibilité cinq, donc en excluant les affectations associées à un niveau de crédibilité inférieur ou égal à deux, représentent un ensemble d'exemples incohérent. En utilisant le modèle MR Sort, il n'y a pas de modèle de préférence qui affecte chacun des dix exemples fournis x_{01} à x_{10} à une des catégories atteignant le niveau de crédibilité deux. C'est pourquoi les affectations aux niveaux de crédibilité cinq et quatre ne peuvent être calculées.

Ces résultats sont présentés à la responsable, qui aimerait disposer de recommandations plus précises. Il est possible de calculer un ensemble de changements minimaux à apporter aux exemples fournis de manière à restaurer la cohérence au niveau de crédibilité cinq (Mousseau *et al.*, 2006, 2003). Modifier l'exemple d'affectation x_{03} des valeurs de crédibilité $\langle 0, 2, 4, 1 \rangle$ aux valeurs $\langle 0, 3, 4, 0 \rangle$ permet de restaurer la cohérence au niveau cinq. La responsable consent à ce changement. Il est alors possible de calculer les affectations robustes des cent étudiants aux niveaux sept, six, et cinq, indiqués en Figure 4.9.

De manière à obtenir des résultats plus précis au niveau de crédibilité le plus bas, la responsable ajoute un nouvel exemple: l'exemple d'affectation a_{004} est ajouté avec les valeurs de crédibilité $\langle 0, 1, 5, 1 \rangle$. Les nouveaux résultats sont présentés dans la figure 4.10. Ces résultats satisfont la responsable.

La responsable choisit d'utiliser les résultats les plus précis, associés au niveau de crédibilité cinq, pour guider l'évaluation des cent étudiants de cette année. Pour certains de ces étudiants, cependant, les catégories proposées à ce niveau de crédibilité ne semblent pas adéquates. Dans ce cas, elle s'autorise à choisir une catégorie parmi celles indiquées par la fonction de tri robuste de crédibilité sept. Elle ne s'autorise pas cependant à sortir de cette recommandation plus large, car elle souhaite s'assurer que sa manière d'évaluer les étudiants présente une cohérence suffisante.

4.5 Conclusion

Ce chapitre propose une procédure pour collecter des informations de préférence du décideur. Les questions posées par cette procédure d'élicitation s'expriment en des termes que le décideur connaît, et non en des termes d'un modèle de préférence opaque pour le décideur. Cela lui permet de réfléchir dans les termes du problème auquel il fait face. Bien que faciles à donner, ces réponses fournissent une information riche. La procédure d'exploitation que nous proposons permet au décideur d'examiner les compromis possibles entre crédibilité et précision lorsqu'il considère les suggestions d'affectations d'un nouvel ensemble d'alternatives.

4 Crédibilités valuées et exploitation simple

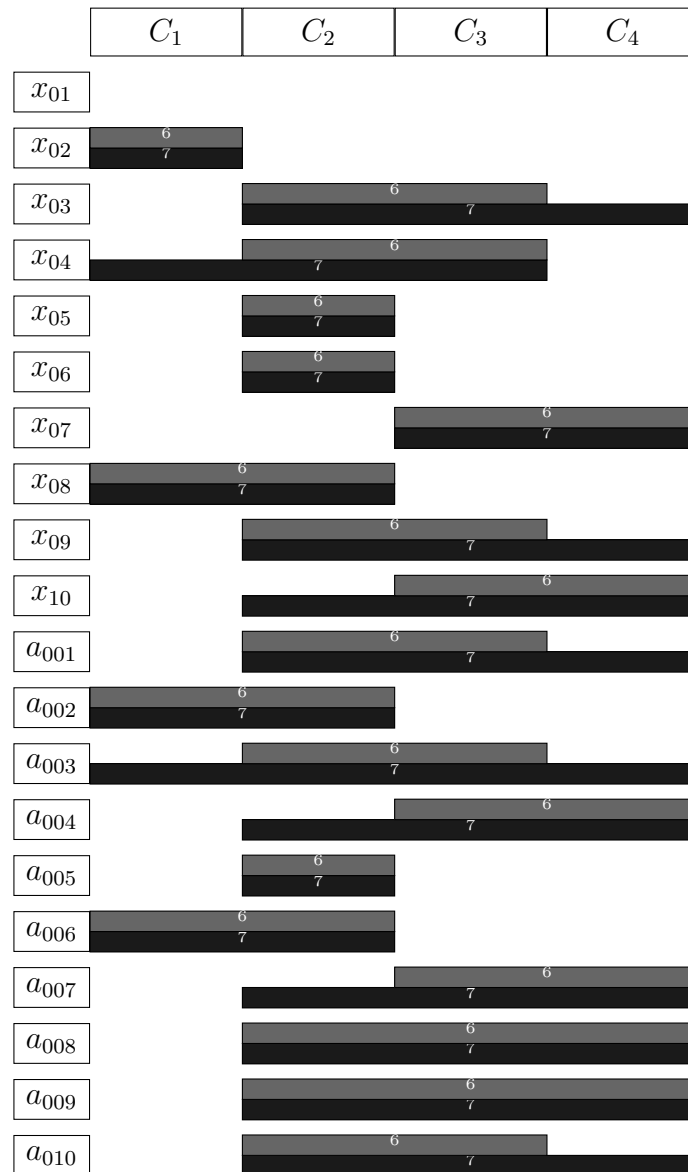


FIGURE 4.8 – Résultats de la procédure. Les niveaux de crédibilités 7 et 6 sont affichés, le niveau de crédibilité 5 ne peut pas être calculé.

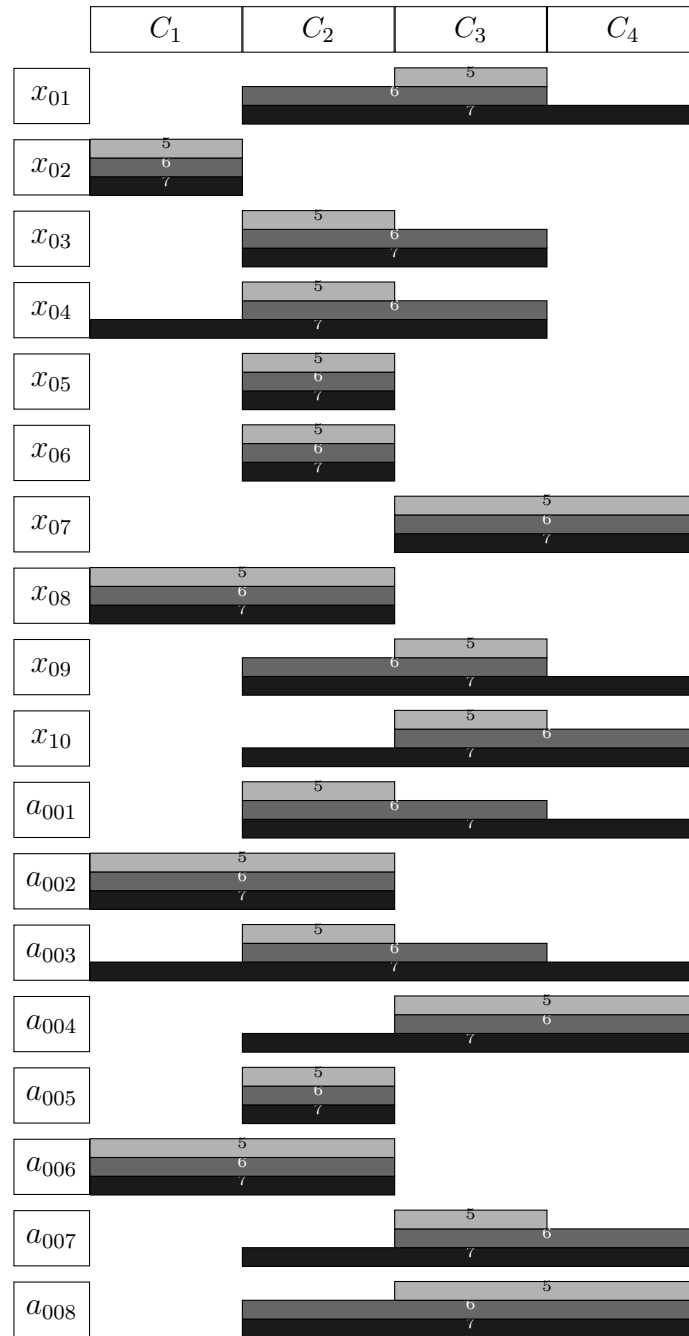


FIGURE 4.9 – Résultats de la procédure, deuxième itération.

4 Crédibilités évaluées et exploitation simple

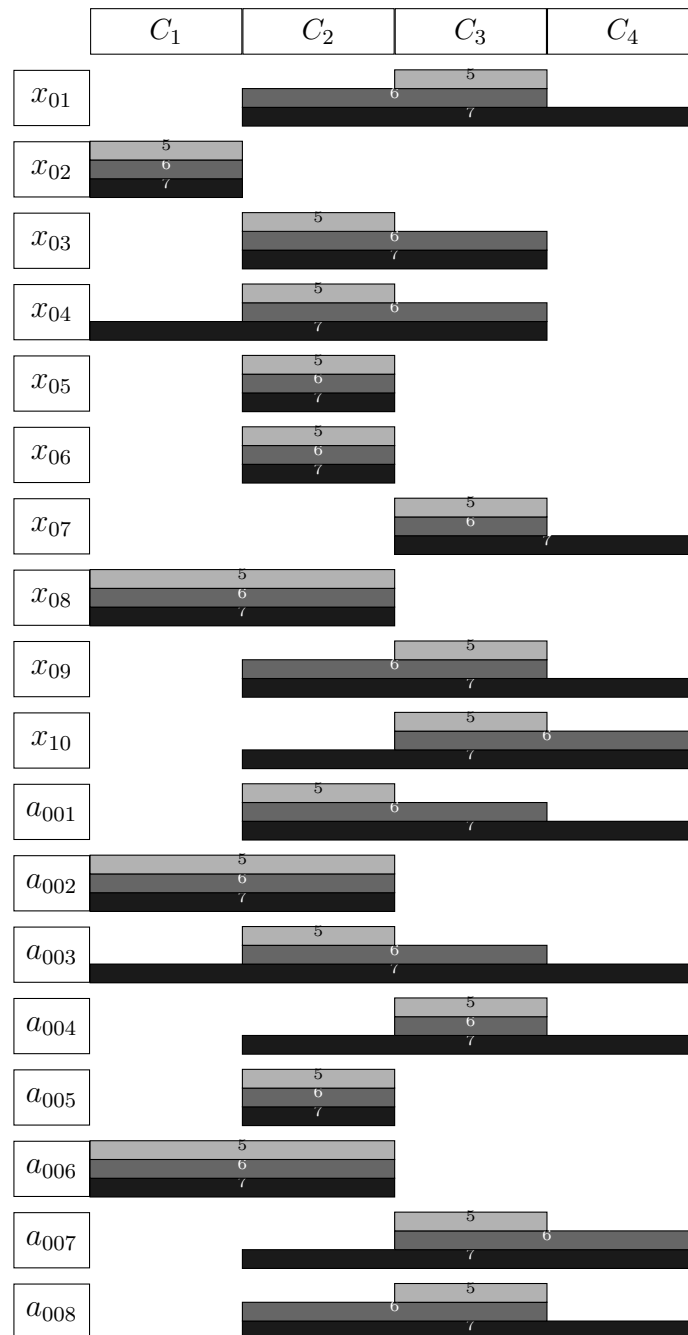


FIGURE 4.10 – Résultats de la procédure, troisième itération.

Ceci ouvre de nouvelles pistes de recherche. D'autres procédures d'exploitation à base d'exemples d'affectation valués peuvent être proposées. Il serait intéressant d'étudier les procédures possibles sur une base axiomatique.

Ce travail ouvre la porte à une meilleure aide à la décision pour un groupe de décideurs. En effet, si de multiples décideurs fournissent une information sous forme d'exemples d'affectation valués, cela permet de chercher des modèles de préférence consensuels, bien qu'éventuellement associés à une faible crédibilité pour tous les décideurs, ou des modèles de préférence associés à une forte crédibilité pour une majorité de décideurs mais laissant certains décideurs insatisfaits.

5 Une méthode de sélection de portefeuilles équilibrés

Dans ce chapitre, nous proposons une procédure de choix de portefeuille dans un contexte multicritère. La procédure permet de sélectionner un portefeuille qui respecte des contraintes d'équilibre, par exemple une bonne balance entre les genres dans un problème de sélection d'étudiants. De plus, l'élicitation des préférences est intégrée à la procédure de choix de portefeuille, ce qui permet de trouver un équilibre entre la sélection des alternatives les meilleures possibles, considérées individuellement, et la satisfaction de desiderata liés à la qualité du groupe. La procédure a été publiée dans la série Lecture Notes in Computer Science (LNCS) (Zheng *et al.*, 2011), et a été utilisée dans une application réelle de sélection d'étudiants (Le Cardinal *et al.*, 2011).

5.1 Introduction à la sélection de portefeuilles

Le décideur est parfois intéressé par la sélection d'un ensemble cohérent d'alternatives, appelé portefeuille. Une part des recherches dans ce domaine vient du monde de la finance, où il s'agit de sélectionner un portefeuille d'actions parmi un ensemble d'actions possibles. Le portefeuille doit habituellement contenir des actions plus ou moins risquées, ayant des sources de risques différentes, afin de diversifier le risque. Cela requiert d'éliciter la propension du décideur à apprécier les investissements risqués. Le problème du choix de portefeuille a été également largement traité dans des contextes plus généraux que la finance. Souvent, le portefeuille doit obéir à des contraintes provenant de la limitation des ressources disponibles. Un exemple typique est la sélection d'un ensemble de projets à financer dans la limite d'un budget donné, pour un organisme visant à choisir des investissements. Lorsqu'il y a plus d'une ressource limitée à considérer, cela conduit à devoir résoudre un problème de sac-à-dos multi-objectif : trouver un sous-ensemble d'alternatives ayant la plus grande valeur totale sous contrainte que les alternatives sélectionnées (mises dans le sac-à-dos, dans la métaphore) consomment ensemble un nombre de ressources inférieur à la limite des ressources disponibles. Dans la métaphore, une des ressources limitées est le poids total toléré dans le sac-à-dos, ou le

volume total disponible.

Dans un contexte de sélection de portefeuilles multicritère, les alternatives parmi lesquelles choisir sont évaluées sur plusieurs critères, et un des aspects du problème est d'éliciter les préférences du décideur, c'est-à-dire de mesurer comment le décideur envisage les compromis entre différents critères conduisant à une évaluation globale des alternatives.

Les recherches proposant des méthodes de sélection de portefeuilles sont extrêmement vastes, et il est illusoire d'en faire un compte-rendu fidèle ici. Les méthodes de sélection de portefeuilles suivent généralement le canevas suivant (Phillips et Bana e Costa (2007) donnent un bon aperçu des approches habituelles, dont nous nous inspirons ici). Chaque alternative est associée à une valeur globale, souvent à l'aide d'une méthode bien connue en finance, en calculant la Net Present Value (NPV) de chaque alternative (en général une alternative est dans ce contexte un projet potentiel qu'il faut choisir de financer ou non). Pour gérer l'aspect multicritère, une approche courante est de traduire tous les aspects en termes monétaires, par exemple en utilisant le « *willingness to pay* ». La méthode d'aide à la décision MAVT est une alternative à l'utilisation de NPV. Le risque est parfois intégré à de telles analyses comme un critère parmi les autres. D'autres méthodes proposent de prendre le risque en compte dans le calcul de la NPV en le modélisant à l'aide de la théorie des probabilités.

Dans le cas le plus simple, il suffit ensuite de sélectionner chaque alternative (financer chaque projet) dont la valeur donnée par NPV est positive, car une telle valeur indique un bénéfice attendu moyen supérieur aux coûts de financement. Une autre possibilité est de sélectionner les projets amenant la plus grande valeur globale sans dépasser les ressources disponibles, en résolvant un problème de sac-à-dos multi-objectif.

5.2 Contexte du travail

Le type de problème qui nous intéresse ici considère deux aspects dans le choix du meilleur portefeuille. Le premier, classique en aide multicritère à la décision, est l'évaluation globale de chaque alternative sur base de ses performances par critère : est-elle globalement suffisamment bonne pour mériter d'être prise dans le portefeuille ? Mais trouver le meilleur portefeuille ne consiste pas nécessairement à prendre les meilleures alternatives jusqu'à épuisement du budget, ou plus généralement des contraintes de ressources. Nous considérons également un deuxième aspect, l'évaluation de la qualité du portefeuille dans son ensemble. Il s'agit là de contraintes plus générales que les contraintes de ressources, telles que des contraintes d'équilibre. Dans un pro-

blème de financement, le décideur peut souhaiter financer le plus également possible différents départements de son entreprise, ou différentes régions sous son influence s'il s'agit d'un responsable politique. Dans un problème d'allocation de ressources à des personnes, le décideur peut préférer un portefeuille équilibré en termes de catégories socio-économiques.

5.2.1 Exemples d'applications

Considérons une université qui doit chaque année sélectionner les étudiants admis à intégrer l'université. L'université désire admettre les étudiants ayant de bonnes notes, et plus généralement de bonnes performances sur les critères d'évaluation considérés (leur motivation, par exemple, telle que mesurée lors d'un entretien). Mais le groupe d'étudiants sélectionné doit également satisfaire à certaines conditions exprimées sur le groupe. Par exemple, le genre des étudiants doit être équilibré, ou il faut une proportion minimale de représentants de certains groupes socio-économiques. De plus, le nombre de places disponibles est restreint. Il faut donc choisir un nombre limité d'étudiants satisfaisant au mieux les contraintes de qualité individuelle et les contraintes d'équilibre du groupe. L'université souhaite également mettre en place une liste d'attente, sur laquelle peuvent être inscrits les étudiants suffisamment bons individuellement mais non sélectionnés dans le groupe d'admis.

Un autre exemple de sélection de portefeuilles de ce type est le problème d'allocation de budget à des projets de recherche. Le comité évalue les mérites des différentes propositions sur les aspects d'originalité, de rigueur, mais souhaite également s'assurer que le groupe de projets finalement retenus est correctement panaché entre les disciplines de recherche, institutions, voire régions. Le groupe de projets retenus doit également être compatible avec le budget disponible.

Ces deux problèmes impliquent d'évaluer des alternatives individuellement, en fonction de leurs performances sur divers critères. Ils ont également en commun de considérer un portefeuille comme adéquat non seulement parce qu'il contient des alternatives jugées intrinsèquement bonnes, mais parce qu'il satisfait à certaines contraintes exprimées sur le groupe.

5.2.2 Travaux proches

Peu de recherches apportent une attention particulière au choix de portefeuilles équilibrés. Citons quelques travaux typiques de cette littérature. Nous reviendrons après avoir exposé notre approche sur ce qui la distingue des propositions existantes.

Farquhar et Rao (1976) établissent une mesure de distance sur chaque attribut, servant à mesurer la dispersion des performances sur chaque attribut pour un portefeuille donné. Le portefeuille retenu est celui qui apporte la plus grande valeur totale, la mesure de dispersion intervenant dans le calcul de cette valeur totale. La fonction attribuant une valeur à un portefeuille utilise des poids associés aux différents attributs, qui sont déterminés par inférence à partir de comparaisons de différents portefeuilles par le décideur. Leur modèle ne comprend pas de notion d'évaluation absolue, bien qu'il puisse être facilement étendu pour ajouter une phase initiale de rejet des alternatives trop mauvaises (Rao *et al.*, 1991).

Golabi *et al.* (1981) utilisent quand à eux, plus classiquement, une fonction de valeur sur les alternatives composant le portefeuille, plutôt que sur le portefeuille. Ils proposent d'utiliser, en plus des contraintes classiques de budget à respecter, des contraintes pour éliminer les portefeuilles insuffisamment équilibrés : le portefeuille choisi est donc celui qui maximise la somme des valeurs des alternatives sélectionnées en respectant les contraintes de budget et les contraintes d'équilibre. La fonction de valeur sur les alternatives est établie par élicitation directe classique des fonctions de valeurs partielles (Keeney et Raiffa, 1976).

Des travaux récents (Liesiö *et al.*, 2007, 2008) appliquent des méthodes d'analyse de préférence robuste à la sélection de portefeuilles. Dans un cas où les informations en présence sont incomplètes, par exemple parce que les poids sont partiellement définis à l'aide de contraintes ou que les performances ne sont pas toutes connues, ils définissent des mesures permettant d'éclairer le décideur, par exemple, sur les alternatives qui seront nécessairement choisies, quelle que soit la façon dont les informations partielles sont précisées. Ils proposent également des algorithmes permettant de traiter ces questions.

5.2.3 Définition du portefeuille par une fonction de tri

La définition de portefeuille utilisée dans ce chapitre diffère quelque peu de la définition usuelle. Nous définissons le problème du choix de portefeuille, étant donné un ensemble d'alternatives A , comme la création d'une partition à k ensembles, ordonnés par préférence, sur l'ensemble A . La partition créée est couvrante, et ses parties disjointes : en notant A_1, \dots, A_k les sous-ensembles créés, on a $\bigcup A_1, \dots, A_k = A$ et $A_i \cap A_j = \emptyset, \forall i, j \in \{1, \dots, k\}$. Certains sous-ensembles peuvent être vides. Cette définition permet de faire du problème du choix d'une partition un problème de tri : les alternatives affectées à une catégorie C_i constituent l'ensemble A_i cherché. La littérature fait généralement référence, en parlant d'un problème de sélection de portefeuille, à un problème de sélection d'un sous-ensemble d'alternatives. Ceci est un cas particulier de

notre définition. Il suffit de définir un problème à deux catégories et de sélectionner les alternatives attribuées à la meilleure catégorie.

Ainsi, définir une fonction de tri revient à définir une manière de sélectionner un portefeuille. Nous proposons dans ce chapitre une manière de définir une fonction de tri de type MR Sort afin de résoudre un problème de portefeuille.

5.3 Formulation du programme mathématique

Notre objectif est de trouver les paramètres d'une fonction de tri de type MR Sort apte à définir des portefeuilles respectant les préférences du décideur. Les préférences du décideur sont capturées à deux niveaux distincts, et formulées comme des contraintes d'un PM. Les préférences au niveau individuel, servant à définir ce que le décideur considère comme étant intrinsèquement une bonne alternative, peuvent être définies par des exemples d'affectation. Au niveau du portefeuille, une vaste classe d'informations de préférences peuvent être prises en compte (limitation de ressources, équilibre des alternatives choisies sur un certain attribut, ...) grâce à des contraintes exprimées sur la taille des catégories. La résolution du PM défini par ces contraintes fournit à la fois un modèle de préférence, un portfolio adéquat, et une explication du choix.

Une partie du programme est très similaire au programme ICL (PM 1, page 77), à la différence que le cas traité ici n'implique qu'un décideur. Les exemples d'affectation sont représentés par un seul ensemble E , contenant des paires $(a, [\underline{C}, \overline{C}])$ indiquant qu'une alternative doit être affectée dans un intervalle de catégories donné. Un tel exemple est noté $(a \rightarrow [\underline{C}, \overline{C}])$. L'ensemble des alternatives utilisées comme exemples est noté X , l'ensemble des alternatives à trier est noté A . Les variables utilisées sont les suivantes.

Performance des profils $l_j, \forall l \in L, j \in \mathcal{J}$: la performance, sur le critère j , du profil l . Cette variable est continue.

Poids $w_j, \forall j \in \mathcal{J}$: le poids du critère j . Il s'agit d'une variable continue non négative.

Seuils de majorité λ : le seuil de majorité, variable continue dans $[1/2, 1]$.

Concordance partielle binaire $K_j(a, l^-(C)), \forall a \in X, C \in \mathcal{C} \setminus C_1, j \in \mathcal{J}$: la concordance partielle, du point de vue du critère j , indiquant si l'alternative a mérite d'atteindre la catégorie C , ou de manière équivalente, indiquant si l'alternative a est au moins aussi bonne que le profil $l^-(C)$. La variable $K_j(a, l^-(C))$ vaut 1 si la performance de l'alternative a sur le critère j est au moins aussi bonne que la performance du profil $l^-(C)$, 0 sinon. Elle est définie comme une variable binaire du PM.

Concordance partielle pondérée $\Psi_j(a, l^-(C)), \forall a \in X, C \in \mathcal{C} \setminus C_1, j \in \mathcal{J}$: une variable équivalente à la variable de concordance partielle binaire, mais pondérée par le poids w_j . Sa valeur est $\Psi_j(a, l^-(C)) = w_j K_j(a, l^-(C))$. Cette variable vaut donc w_j si la performance de l'alternative a sur le critère j est au moins aussi bonne que la performance du profil $l^-(C)$, 0 sinon. C'est une variable continue non négative.

Affectation $n(a, C), \forall a \in A, C \in \mathcal{C}$ est une variable binaire indiquant si l'alternative a est affectée à la catégorie C . Ces variables sont nécessaires pour définir les contraintes sur le portefeuille.

Nous ne décrivons pas les contraintes assurant le respect des exemples d'affectation car elles sont simplement l'équivalent mono-décideur de celles utilisées dans ICL. La nouveauté par rapport à ICL concerne les contraintes sur le portefeuille.

Dans cette section, notons $(a \rightarrow C)$ pour indiquer que l'alternative a , parmi les alternatives à classer A , est affectée à la catégorie C par un modèle de préférence donné.

$$(a \rightarrow C) \Leftrightarrow \begin{cases} \sum_{j \in \mathcal{J}} \Psi_j(a, l^-(C)) \geq \lambda \text{ et} \\ \sum_{j \in \mathcal{J}} \Psi_j(a, l^+(C)) < \lambda. \end{cases}$$

Supposons qu'un décideur souhaite imposer, dans un problème de sélection d'étudiants, qu'au moins trente étudiants dans la meilleure catégorie (C_k) soient des femmes. Pour modéliser cela, définissons une fonction Genre sur l'ensemble des alternatives $a \in A$, qui vaut un si l'étudiant a est une femme, et zéro sinon. Il suffit ensuite de définir une contrainte exigeant que la somme des Genre(a) sur chaque alternative a affectée à C_k atteigne au moins trente : $\sum_{(a \rightarrow C_k)} \text{Genre}(a) \geq 30$.

Dans un problème de sélection de projets, supposons que le décideur souhaite s'assurer que la somme des couts des projets sélectionnés (affectés dans la meilleure catégorie C_k) n'excède pas le budget disponible x . Une fonction Cout peut être définie sur l'ensemble des alternatives représentant leur cout, et la contrainte suivante peut être définie : $\sum_{(a \rightarrow C_k)} \text{Cout}(a) \leq x$.

Représentons les préférences exprimées par le décideur concernant le portefeuille par un ensemble N de tuples $\langle C, \underline{\eta}, \bar{\eta}, P \rangle, C \in \mathcal{C}, \underline{\eta}, \bar{\eta} \in \mathbb{R}$, où P désigne une fonction de A vers \mathbb{R} . Un tel tuple indique que le nombre d'alternatives de A affectées à C , pondéré par l'attribut P , doit valoir au moins $\underline{\eta}$ et au plus $\bar{\eta}$: $\underline{\eta} \leq \sum_{(a \rightarrow C)} P(a) \leq \bar{\eta}$.

Afin d'implémenter cette contrainte, définissons des variables $n(a, C)$ binaires, $\forall a \in A, C \in \mathcal{C}$, telles que $n(a, C)$ égale 1 ssi a est affectée à la catégorie

C . Il faut garantir :

$$n(a, C) = 1 \Leftrightarrow (a \rightarrow C).$$

Ceci peut être fait grâce aux contraintes linéaires suivantes, $\forall a \in A, C \in \mathcal{C}$:

$$\left\{ \begin{array}{l} \sum_{j \in J} \Psi_j(a, l^-(C)) \geq \lambda + (n(a, C) - 1), \\ \sum_{j \in J} \Psi_j(a, l^+(C)) < \lambda + (1 - n(a, C)), \end{array} \right. \quad (n \text{ implique affectation})$$

et, $\forall a \in A$:

$$\sum_{C \in \mathcal{C}} n(a, C) = 1. \quad (\text{somme } n)$$

Les contraintes (n implique affectation) garantissent que $n(a, C) = 1 \Rightarrow (a \rightarrow C)$. La contrainte (somme n) fait en sorte, pour chaque alternative a , qu'exactement une des variables $n(a, C)$ parmi toutes les catégories C égale 1.

Une fois les variables $n(a, C)$ définies, les contraintes principales peuvent être ajoutées, $\forall \langle C, \underline{\eta}, \bar{\eta}, P \rangle \in N$:

$$\underline{\eta} \leq \sum_{a \in A} n(a, C)P(a) \leq \bar{\eta}. \quad (\text{taille catégorie})$$

L'ensemble des contraintes définissant le programme est repris dans le PM 3.

Comme pour ICL, la fonction objectif peut être de maximiser une variable slack s de distance entre les supports et le seuil de majorité. On peut également simplement chercher une solution satisfaisant toutes les contraintes.

Notons que les préférences indiquées par le décideur ne définiront très probablement pas une solution unique. Soit les préférences sont trop contraignantes, et aucun portefeuille possible ne satisfera toutes les contraintes, soit elles définiront un ensemble de portefeuilles compatibles. L'approche que nous proposons devrait être utilisée de façon interactive, de manière à ce que le décideur puisse raffiner ou relâcher les contraintes représentant ses préférences jusqu'à trouver un portefeuille qui le satisfasse.

Le programme correspondant, ainsi que des variantes qui peuvent être construites à partir de ce programme, est implémenté dans la bibliothèque J-MCDA (cf. Section 6.2).

5.4 Exemple illustratif

Pour illustrer la méthode, reprenons le contexte de l'exemple traité en section 3.1.6. Un comité gouvernemental doit choisir quels projets de recherche

5 Une méthode de sélection de portefeuilles équilibrés

Trouver une solution satisfaisant

$$\begin{aligned}
 & \sum_{j \in \mathcal{J}} w_j = 1. \\
 \forall C \in \mathcal{C} \setminus \{C_1, C_n\}, j \in \mathcal{J} : & \quad l_j^-(C) \leq l_j^+(C). \\
 \forall a \in X, l \in L, j \in \mathcal{J} : & \quad \frac{(a_j - l_j) + \varepsilon}{M} \leq K_j(a, l) \leq \frac{a_j - l_j}{M} + 1. \\
 \forall a \in X, l \in L, j \in \mathcal{J} : & \quad \begin{cases} w_j + K_j(a, l) - 1 \leq \Psi_j(a, l) \leq K_j(a, l) \\ \Psi_j(a, l) \leq w_j. \end{cases} \\
 \forall (a \rightarrow [\underline{C}, \overline{C}]) \in E, \underline{C} \neq C_1 : & \quad \sum_{j \in \mathcal{J}} \Psi_j(a, l^-(\underline{C})) \geq \lambda. \\
 \forall (a \rightarrow [\underline{C}, \overline{C}]) \in E, \overline{C} \neq C_n : & \quad \sum_{j \in \mathcal{J}} \Psi_j(a, l^+(\overline{C})) \leq \lambda - \varepsilon. \\
 \forall a \in A, C \in \mathcal{C} : & \quad \begin{cases} \sum_{j \in \mathcal{J}} \Psi_j(a, l^-(C)) \geq \lambda + (n(a, C) - 1), \\ \sum_{j \in \mathcal{J}} \Psi_j(a, l^+(C)) \leq \lambda + (1 - n(a, C)) - \varepsilon. \end{cases} \\
 \forall a \in A : & \quad \sum_{C \in \mathcal{C}} n(a, C) = 1. \\
 \forall \langle C, \underline{\eta}, \overline{\eta}, P \rangle \in N : & \quad \underline{\eta} \leq \sum_{a \in A} n(a, C) P(a) \leq \overline{\eta}.
 \end{aligned}$$

PM 3: Contraintes pour le choix d'un portefeuille.

financer parmi une liste de cent propositions. Pour ce faire, il souhaite trier ces propositions dans trois catégories : les projets excellents, qui doivent être financés (catégorie *Excellent*), les projets de bonne qualité, à financer si un budget suffisant peut être trouvé (catégorie *Bon*), et les projets de qualité insuffisante, qui ne méritent pas de financement (catégorie *Mauvais*). Les six critères suivants sont utilisés.

- sq** La qualité scientifique du projet, évaluée sur une échelle ordinaire à cinq points.
- wq** La qualité de rédaction, évaluée sur une échelle ordinaire à cinq points.
- ad** L'adéquation entre le projet et les priorités du gouvernement, évaluée sur une échelle ordinaire à trois points.
- te** L'expérience des équipes de recherche associées au projet, évaluée sur une échelle ordinaire à cinq points.

Projet	Critères d'évaluation						Attributs de description		
	rq	ps	a	sq	te	ic	budget	domaine	pays
Pr 001	2	47	2	3	1	0	27	Stat.	Allemagne
Pr 002	2	3	2	4	4	0	29	Stat.	France
Pr 003	5	63	1	5	1	0	20	Stat.	Italie
Pr 004	1	92	3	5	5	1	34	AI	Allemagne
Pr 005	4	13	2	4	2	0	32	Stat.	Allemagne
Pr 006	5	5	3	5	1	0	22	Stat.	Pays-Bas
Pr 007	1	27	3	2	5	1	34	OR	Allemagne
⋮									

TABLE 5.1 – Une partie des projets de recherche à évaluer. Le budget est en dizaine de milliers d'euros.

ic Une indication binaire indiquant si le projet inclut une collaboration internationale.

ps La valeur agrégée des publications des chercheurs participant au projet, évaluée par un nombre entre zéro et cent.

En plus de ces six critères, les cent projets à évaluer sont décrits par trois attributs : le domaine de recherche du projet (Recherche opérationnelle (OR), Intelligence artificielle (AI) ou Statistiques), le budget demandé pour le projet, le pays d'origine. La table 5.1 reprend les données concernant les sept premiers projets de la liste. Les données utilisées dans cet exemple sont disponibles en format électronique (voir annexe D, numéro 4).

De manière à indiquer leur façon d'évaluer la qualité intrinsèque des projets, le comité donne trente exemples de projets de recherche rencontrés précédemment, dont les performances sur les six critères et les évaluations finales sont connues. Une partie de ces données est indiquée en Table 5.2. Les exemples utilisés ici sont précis, donc n'utilisent pas d'intervalles.

Le programme d'inférence est utilisé avec ces exemples d'affectation, sans y ajouter de contraintes de portefeuille. La table 5.3 montre les profils et coalitions résultants. Ce modèle de préférence est alors utilisé pour évaluer les cent projets de recherche, ce qui conduit à évaluer vingt-deux projets comme excellents (catégorie *Excellent*).

Le comité n'est pas satisfait par cet ensemble de projets, car il requiert un budget de 718, dépassant largement le budget disponible de 400. Le programme est donc lancé à nouveau, avec une contrainte limitant la somme des couts des projets affectés à la catégorie *Excellent*. Cette deuxième exécution conduit à

5 Une méthode de sélection de portefeuilles équilibrés

Projet	rq	ps	a	sq	te	ic	Catégorie
Ex 01	4	50	2	3	3	0	Bon
Ex 02	4	85	3	1	5	1	Excellent
Ex 03	3	95	1	2	5	1	Bon
Ex 04	5	91	2	2	5	1	Excellent
Ex 05	5	89	1	5	3	0	Excellent
Ex 06	3	5	3	2	2	1	Bon
⋮							

TABLE 5.2 – Certains des projets utilisés en exemples et leurs affectations respectives.

	rq	ps	a	sq	te	ic
$l^-(C_3)$	4	96	4	5	3	1
$l^-(C_2)$	2	73	4	1	2	1
w	0,2	0,2	0	0,2	0,2	0,2
$\lambda = 0,5$						

TABLE 5.3 – Profils et coalitions inférés lors de la première itération.

un modèle de préférence indiqué à la table 5.4, et à une nouvelle affectation des cent projets dont une partie est affichée en Table 5.5. Onze projets sont affectés à la catégorie *Excellent*, représentant une charge budgétaire totale qui est effectivement en-dessous des 400.

Cependant, le comité trouve cette proposition trop déséquilibrée quand aux domaines de recherches financés. Le domaine AI voit sept projets sélectionnés, tandis que le domaine OR n'a qu'un projet financé. Une troisième itération a donc lieu, incluant une contrainte que chaque domaine de recherche ait au moins deux projets affectés en catégorie *Excellent*. Les affectations finales, que la table 5.6 montre partiellement, sont considérées pleinement satisfaisantes.

	rq	ps	a	sq	te	ic
$l^-(C_3)$	3	84	2	4	3	2
$l^-(C_2)$	2	2	2	1	2	1
w	0,143	0,143	0,143	0,143	0,286	0,143
$\lambda = 0,643$						

TABLE 5.4 – Profils et coalitions inférés avec une contrainte de budget.

5.4 Exemple illustratif

Projet	rq	ps	a	sq	te	ic	budget	domaine	pays	Catégorie
Pr 001	2	47	2	3	1	0	27	Stat.	All.	Mauvais
Pr 002	2	3	2	4	4	0	29	Stat.	Fr.	Bon
Pr 003	5	63	1	5	1	0	20	Stat.	It.	Mauvais
Pr 004	1	92	3	5	5	1	34	AI	All.	Excellent
Pr 005	4	13	2	4	2	0	32	Stat.	All.	Bon
Pr 006	5	5	3	5	1	0	22	Stat.	P.-B.	Mauvais
Pr 007	1	27	3	2	5	1	34	OR	All.	Bon
⋮										

TABLE 5.5 – Une partie des projets de recherche affectés à l’aide du modèle de préférence inféré lors de la deuxième itération.

Projet	rq	ps	a	sq	te	ic	budget	domaine	pays	Catégorie
Pr 001	2	47	2	3	1	0	27	Stat.	All.	Bon
Pr 002	2	3	2	4	4	0	29	Stat.	Fr.	Bon
Pr 003	5	63	1	5	1	0	20	Stat.	It.	Mauvais
Pr 004	1	92	3	5	5	1	34	AI	All.	Excellent
Pr 005	4	13	2	4	2	0	32	Stat.	All.	Bon
Pr 006	5	5	3	5	1	0	22	Stat.	P.-B.	Bon
Pr 007	1	27	3	2	5	1	34	OR	All.	Bon
⋮										

TABLE 5.6 – Une partie des projets de recherche affectés à l’aide du modèle de préférence inféré lors de la troisième itération.

Le processus aurait pu continuer si le comité avait souhaité trouver un meilleur équilibre entre les pays recevant des fonds, ou s'il s'était intéressé de plus près à la catégorie *Bon*. À un certain stade du processus, il aurait été possible que le programme d'inférence ne trouve plus de solution, car des contraintes trop fortes auraient été imposées. Dans ce cas, les algorithmes proposés par Mousseau *et al.* (2006) auraient pu être utilisés pour rechercher un ensemble minimal de contraintes à relâcher afin de restaurer la faisabilité du programme.

La résolution des problèmes illustrés ici prend moins d'une minute sur un ordinateur de bureau classique, avec l'implémentation disponible dans la bibliothèque J-MCDA (cf. Section 6.2).

5.5 Conclusion et perspectives

5.5.1 Caractéristiques distinctives de l'approche

La procédure que nous proposons permet de prendre en compte deux éléments d'appréciation distincts afin de définir ce qu'est un bon portefeuille. Une composante est l'évaluation de la qualité intrinsèque des alternatives. Notre approche permet d'exclure du portefeuille choisi les alternatives de qualité insuffisante, où cette notion de qualité est déterminée de manière absolue, indépendamment donc des autres alternatives. L'autre composante est l'évaluation relative des alternatives : un bon portefeuille est un portefeuille équilibré, ou un portefeuille qui respecte des contraintes définies sur le groupe d'alternatives. Ces deux composantes sont généralement prises en compte par d'autres méthodes de sélection de portefeuilles, mais notre approche se distingue de plusieurs façons.

Un élément central est que l'élicitation des préférences permettant de définir ce qui est jugé comme étant une alternative intrinsèquement bonne est liée à l'élicitation des préférences visant à définir la qualité du portefeuille complet. Les préférences concernant les alternatives individuelles sont définies de façon lâches, à l'aide d'exemples d'affectation, ce qui définit un ensemble de modèles de préférence compatibles et laisse de la latitude pour trouver un portefeuille qui respecte également des contraintes exprimées sur le groupe d'alternatives sélectionné. En une étape, la procédure définit un modèle de préférence précis, et un portefeuille : déterminer le modèle de préférence, c'est déterminer le portefeuille.

Cette approche contraste avec les approches existantes traitant de l'évaluation multicritère intrinsèque et de l'évaluation de la qualité du portefeuille dans son ensemble. Les approches existantes telles celles analysées précédemment

travaillent plutôt en deux étapes, en excluant les alternatives insuffisamment bonnes puis en sélectionnant le portefeuille le plus équilibré, ou inversement.

5.5.2 Une alternative à la discrimination positive

Il est clair que le problème analysé a des liens étroits avec la question de la discrimination positive. Ceci se voit en particulier lorsque le problème de portefeuille consiste à choisir un sous-ensemble d'alternatives équilibré, donc à partitionner en deux catégories avec la sémantique que la meilleure catégorie contient les alternatives retenues, et que les autres sont des personnes.

La discrimination positive (Calvès, 2010) consiste à privilégier le choix d'une personne appartenant à une catégorie de gens habituellement discriminés négativement, dans un souci de rétablir un certain équilibre, en introduisant une inégalité de traitement cette fois favorable à la catégorie habituellement délaissée. Par exemple, dans une université où les étudiantes seraient sous-représentées par rapport aux étudiants, l'université pourrait adopter un niveau d'exigence plus élevé concernant les dossiers des hommes pour rétablir l'équilibre. La discrimination positive est parfois critiquée par ceux qui considèrent qu'il n'est pas légitime de compenser une inégalité en en introduisant une autre, quand bien même celle-ci vise à compenser une inégalité existante. Outre le principe de justice, ses détracteurs font également remarquer qu'elle peut être contre-productive car elle risque de générer de la rancœur auprès des nouvelles victimes qu'elle crée. Cette rancœur risque qui plus est d'aller à l'encontre justement des personnes qu'il s'agissait de favoriser.

L'approche que nous proposons permet au décideur qui le souhaite de chercher une solution équilibrée, donc de réduire la discrimination, mais permet également d'éviter les traitements de faveur. Cela est possible dans certaines circonstances seulement. La sous-représentation d'une catégorie de gens dans un contexte donné peut être due à une importance accordée à certains critères sur lesquels cette catégorie de gens est moins bonne. Cette importance particulière peut ne pas être le reflet d'un choix réfléchi, et il est possible qu'une autre logique existe, qui puisse également être considérée comme raisonnable par le décideur, et qui ne mène pas à la discrimination problématique évoquée. Cet autre mode de raisonnement ne doit pas nécessairement consister en un traitement de faveur envers la catégorie considérée comme victime de la discrimination.

Une propriété importante de notre méthode est qu'elle traite toutes les alternatives de la même façon, une fois le modèle de tri déterminé, sans utiliser les attributs (donc en n'utilisant que les critères). Cela signifie qu'elle garantit que si une alternative est choisie plutôt qu'une autre, c'est qu'il existe une fonction raisonnable qui la préfère en ne considérant que les critères. Raisonnable signi-

fie ici que la fonction est comprise dans la classe de modèles choisie, supposée représenter les fonctions de tri raisonnables aux yeux du décideur. Ainsi, si le sexe de l'étudiant est représenté comme un attribut et non comme un critère, la méthode permet au décideur de chercher une façon raisonnable (représentable par MR Sort) de sélectionner les étudiants en assurant un équilibre adéquat. Le mode de raisonnement adopté par la classe de modèles choisie s'applique alors à la sélection des étudiants : par exemple, il garantit que les alternatives retenues ne sont pas dominées.

D'autres façons de faire ne permettent pas aussi facilement cette recherche. Par exemple, dans l'approche de Golabi *et al.* (1981) examinée précédemment, une fonction de valeurs précise est utilisée pour évaluer les alternatives. Leur méthode ne propose pas de moyen de s'assurer que cette fonction est non discriminatoire. Lors du choix d'un portefeuille donné, l'exclusion des portefeuilles qui ne respectent pas les contraintes d'équilibre pourrait conduire à inclure dans le portefeuille retenu une alternative qui est dominée par une autre. Ceci pourrait arriver alors même qu'une autre fonction de valeurs, moins discriminatoire, aurait pu être utilisée.

5.5.3 Perspectives

Il serait intéressant d'analyser le temps de résolution des problèmes en tenant compte des contraintes de portefeuilles afin d'étudier la facilité de résolution en fonction de la taille des problèmes. Des méthodes de résolution adéquates pourraient sans-doute être développées en parallèle du développement de méthodes de résolutions pour le problème initial (sans les contraintes de portefeuille).

La prise en compte de conclusions robustes, de manière similaire aux travaux proposés par Liesiö *et al.* (2007, 2008) mentionnés ci-dessus, pourrait être intégrée à la méthode de manière à fournir des recommandations au décideur ne dépendant que des préférences effectivement exprimées.

Enfin, le problème auquel nous faisons face peut être vu comme un problème bi-objectif, puisque le portefeuille à choisir doit à la fois être aussi équilibré que possible et contenir les alternatives individuellement aussi bonnes que possibles. Il serait intéressant d'introduire une manière de pondérer ces deux objectifs plus subtile que l'imposition de contraintes que nous utilisons ici.

Troisième partie

Implémentation

6 Contributions sous forme de logiciels

Ce chapitre introduit la plateforme de composants logiciels proposée par le consortium Decision-Deck ainsi que le standard XMCD, décrit les contributions logicielles que nous avons fournies sous forme de services web XMCD, et présente une application de la méthode proposée en Section 3.1 à la sélection de risque, en indiquant comment les outils implémentés peuvent être utilisés pour appliquer la méthode. La première section s’inspire de l’introduction d’un article soumis à Decision Support Systems (DSS) (Cailloux *et al.*, 2012c).

6.1 Introduction

Le champ de l’aide multicritère à la décision propose de nombreuses méthodes d’aide à la décision. Des recherches ont montré que ces méthodes sont applicables en pratique (Wallenius *et al.*, 2008), mais beaucoup d’applications se cantonnent aux modèles suffisamment simples pour être calculés manuellement ou avec un logiciel général. Les méthodes qui proposent leur propre logiciel connaissent un succès particulier (Forman et Gass, 2001), mais beaucoup de méthodes n’ont été développées que de manière théorique et aucun logiciel ne permet leur utilisation facile. Lorsqu’un logiciel doit être développé, il faut souvent recommencer de zéro l’implémentation de fonctionnalités qui ont déjà été implémentées par d’autres, faute d’avoir établi dans la communauté de l’AMCD des modes opératoires permettant la réutilisation de composants logiciels. Nous postulons que l’application limitée de certaines méthodes d’AMCD, en particulier celles dont l’implémentation n’est pas triviale, s’explique pour partie par la difficulté de leur intégration dans un système informatique existant (Tervonen, 2012).

Certaines communautés de recherche sont parvenues à créer des plateformes, langages, ou standards de communication ouverts permettant de faciliter le partage et la réutilisation de code implémentant des méthodes. Le standard Predictive Model Markup Language (PMML) (Guazzelli *et al.*, 2009) et la plateforme Weka sont utilisés en Analyse de Données (*Data Mining*). Le langage R a été développé principalement en vue d’implémenter des méthodes d’analyse

statistique. Le standard OS a été proposé pour représenter des PMs de manière à pouvoir les diffuser à travers une plateforme de calcul distribué (Fourer *et al.*, 2009, 2010b,a). Jusqu'à récemment, la communauté de l'AMCD ne proposait pas d'outils comparables.

De nombreux cadres logiciels pour l'AMCD et systèmes de support à la décision (*decision support systems*) implémentant des méthodes d'AMCD ont été proposés (Fedorowicz et Williams, 1986 ; Martin et Fuerst, 1984 ; Minch et Sanders, 1986 ; Georgopoulou *et al.*, 1998 ; Natividade-Jesus *et al.*, 2007 ; Zopounidis et Doumpos, 2000 ; Jármai, 1989 ; Spengler *et al.*, 1998 ; Jiménez *et al.*, 2006 ; van Valkenhoef *et al.*, 2012 ; Gauthier et Néel, 1996). Mais ces systèmes ont souvent été développés en vue d'un objectif spécifique, ou pour un domaine particulier, et ne proposent pas un système général pour réutiliser des composants, ou d'interface de communications avec les composants tiers.

Un intérêt théorique existe concernant la réunion de plusieurs méthodes d'AMCD dans un cadre logiciel commun. Déjà en 1989, Teghem *et al.* (1989) ont analysé plusieurs méthodes d'AMCD et étudié les propriétés qu'elles satisfont, en vue d'en déduire un arbre de décision permettant de sélectionner la méthode la plus appropriée. De même, Hong et Vogel (1991) ont proposé une taxonomie de plusieurs méthodes d'AMCD et ont proposé un système d'enchaînement de règles de décisions basées sur des méthodes différentes. Ce type de propositions requiert un système standard de communication entre composants pour être implémenté de façon plus générale, c'est-à-dire de façon non limitée aux méthodes envisagées par les auteurs de ces propositions.

Le travail de développement de cette thèse s'inscrit pleinement dans un effort entrepris depuis plusieurs années par le consortium Decision Deck. Ce consortium vise à fournir des outils open source implémentant des méthodes d'AMCD. Le consortium a proposé une plateforme qui regroupe des composants open source implémentant chacun une fonctionnalité précise d'une méthode. Comme de nombreuses méthodes présentent des parties communes, rendre disponible des composants individuels permet de réduire le travail nécessaire à l'implémentation d'une nouvelle méthode. Le consortium gère des serveurs hébergeant ces composants et les rend disponibles en tant que services web.

Il est important que de tels composants échangent leurs données dans un format standard, pour qu'ils puissent communiquer. Le consortium a proposé un tel format de données, basé sur XML, nommé XMCDa.

6.1.1 Le format XMCDa

Le format XMCDa a été développé en vue d'établir une façon standard d'encoder des données liées à une méthode d'AMCD ou à un problème de décision. XMCDa (<http://www.decision-deck.org/xmcd/>) (Bigaret et Meyer,

2012) décrit formellement l'encodage requis pour représenter dans un fichier informatique des notions telles que alternative, critère, une performance d'une alternative sur un critère, etc. XMCDA n'est pas restreint à une famille de procédures d'AMCD donnée.

Le format XMCDA est formellement spécifié sous la forme d'un schéma XML (<http://www.w3.org/TR/2004/REC-xmlschema-0-20041028/>). Sans entrer dans les détails, il s'agit d'un fichier informatique conforme à une certaine grammaire, décrit dans le langage XML Schema, qui permet de définir d'autres grammaires.

Notons pour être précis que les standards relatifs à XML dont nous parlons ici, incluant le standard XMCDA, ne se restreignent pas à décrire l'encodage dans des « fichiers », mais s'applique à tout ce qui peut être représenté comme une suite de bits, par exemple un flux dans un réseau. Nous utilisons le terme fichier dans cette partie par simplicité. Nous parlerons d'un fichier conforme à XMCDA ou simplement d'un fichier XMCDA pour désigner un flux de données se conformant à la grammaire XMCDA.

Nous avons proposé une évolution du schéma XMCDA permettant d'éviter quelques inconvénients de la version actuelle, que nous présenterons au chapitre 7. Les services web présentés dans ce chapitre utilisent le schéma XMCDA standard actuel, par souci de compatibilité avec les logiciels existants.

6.1.2 Les services web XMCDA

Les composants mis à disposition par le consortium Decision Deck sont donc appelés des services web XMCDA. Une petite centaine de services web XMCDA sont actuellement publiés (<http://www.decision-deck.org/ws>).

Le consortium propose également un logiciel, *diviz* (Bigaret et Meyer, 2012), qui permet d'accéder à travers une interface graphique aux services web XMCDA. L'utilisateur peut créer son propre enchaînement de composants (pourvu que les entrées et sorties soient compatibles) en les agençant de manière parallèle ou séquentielle à sa guise, et dispose également de services de visualisation qui créent par exemple des graphiques en HTML ou génèrent des images, ce qui permet de voir les résultats des calculs de manière plus agréable qu'en lisant du XML.

En s'appuyant sur des outils génériques implémentés dans notre bibliothèque J-MCDA, nous avons soumis plusieurs services web XMCDA, qui sont actuellement déployés sur les serveurs du consortium. Certains de ces web services remportent la palme des services les plus utilisés parmi les composants de calculs, c'est-à-dire en excluant les services utilisés pour visualiser des résultats (Patrick Meyer, communication personnelle).

Un ensemble de composants permettent, en les assemblant de différentes manières, de reproduire les relations de surclassement calculées par plusieurs méthodes des familles ÉLECTRE et PROMÉTHÉE, présentées en section 2.2. Ces services ont été développés grâce au support de l'Université Libre de Bruxelles et en particulier de Yves De Smet, que l'auteur remercie particulièrement. Ces services web ont fait l'objet d'une publication à la conférence *Uncertainty and Robustness in Planning and Decision Making (URPDM)* (Cailloux, 2010).

Un composant supplémentaire implémente une partie des programmes de désagrégation proposés dans cette thèse.

6.2 Bibliothèques J-MCDA et JLP

Nous avons développé des fonctionnalités génériques utiles au développement d'outils d'AMCD dans une bibliothèque libre et open source, J-MCDA (<http://sourceforge.net/projects/j-mcda/>). Les services web présentés dans ce chapitre ont été développés en prenant appui sur certaines de ces fonctionnalités génériques et représentent un exemple de ce qu'elles permettent.

J-MCDA fournit un ensemble d'objets Java représentant les concepts requis pour manipuler des procédures d'AMCD, tels que des alternatives, des échelles de critères, des matrices de performances, des ensembles de coalitions...

Les fonctionnalités comprennent la lecture et l'écriture au format XMCDA, et au format que nous avons proposé (cf. Chapitre 7). Un cadre général est proposé qui permet de publier facilement des composants XMCDA au format requis par la plateforme proposée par Decision Deck. Entre autres, le cadre fournit la sérialisation et dé-sérialisation automatique des fichiers d'entrée et de sortie. Cela permet au programmeur de se concentrer sur les fonctionnalités qu'il souhaite intégrer à son composant et non sur le code nécessaire pour intégrer son composant dans le cadre logiciel de Decision Deck. La bibliothèque permet d'écrire facilement des convertisseurs. Des exemples sont inclus permettant de convertir des fichiers XMCDA en un format CSV classique (Comma Separated Values) et en un format utilisé par le logiciel IRIS (Dias et Mousseau, 2003).

Le but n'était pas seulement d'implémenter les outils présentés ici, mais de fournir une base réutilisable par d'autres développeurs. Les classes développées satisfont les bonnes pratiques de la conception orientée objets : les détails d'implémentation sont cachés, les méthodes Java présentent un contrat clair, l'implémentation réutilise des bibliothèques tierces parties classiques et bien conçues. Les tests unitaires sont fournis dans le format JUnit (Massol et Husted, 2003). Un ensemble de fichiers exemples au format XMCDA et au format que nous proposons est inclus dans la bibliothèque. Les tests s'appuient sur

ces fichiers exemples. Ils ont également une valeur en soi pour documenter les formats utilisés par la bibliothèque.

Par ailleurs, une part importante des méthodes proposées s'implémente sous la forme de programmes mathématiques. Pour faciliter l'implémentation, nous avons également développé la bibliothèque JLP (Java Linear Programming, <http://code.google.com/p/jlp-interface/>), qui permet de créer et résoudre des PMs depuis le langage Java. La bibliothèque n'est pas liée à l'AMCD, elle a une portée générale et pourrait intéresser tout développeur de PMs. Elle ne contient pas elle-même de solveur, car de nombreux solveurs de qualité existent. Sa valeur est double. D'une part, elle permet de rendre le code plus clair et facile à développer, car elle propose une interface orientée objet simple pour définir et manipuler les PMs et les solutions. D'autre part, elle permet de rendre le code indépendant du solveur : une fois que le code nécessaire à la définition d'un PM est écrit, il suffit de changer le solveur avec lequel JLP s'interface, sans devoir modifier ce code, pour changer le solveur avec lequel le programme est résolu. Cela permet de tester la résolution avec différents solveurs facilement, et cela permet de distribuer son programme à différents types d'utilisateurs sans devoir multiplier les implémentations : l'utilisateur peut au choix invoquer un solveur tel que Cplex, performant mais cher pour les utilisateurs non académiques, ou un solveur gratuit mais potentiellement moins performant. Notons que la fonction d'interchangeabilité automatique du solveur est encore en cours d'écriture à l'heure où ces lignes sont écrites.

6.3 Composants de calcul d'une relation de surclassement

Cette section décrit une série de composants implémentant le calcul de la relation de surclassement des modèles ÉLECTRE IV, ÉLECTRE II, ÉLECTRE III, ÉLECTRE TRI, et le calcul des flux proposé par le modèle PROMÉTHÉE. Cette section doit être lue en conjonction avec la section 2.2 qui décrit ces modèles. La figure 6.1 montre deux de ces composants dans le logiciel diviz.

6.3.1 Concordance

Le composant *ElectreConcordance* calcule une relation de concordance floue ou binaire. La relation calculée correspond à $\Psi(a, b)$, définie à la section 2.2.

Il utilise un ensemble d'alternatives, un ensemble de critères munis de poids, un ensemble de valeurs de performances, et éventuellement un seuil de préférence et un seuil d'indifférence.

6 Contributions sous forme de logiciels

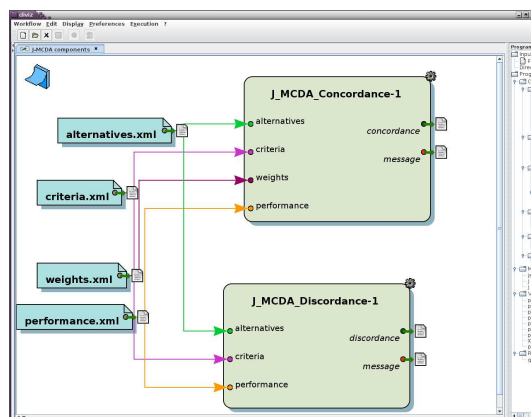


FIGURE 6.1 – Utilisation dans diviz de deux composants de la bibliothèque J-MCDA.

Plus précisément, le composant reçoit un ensemble d'alternatives \mathbb{A} et un ensemble de critères \mathcal{J} , représentés simplement par des identifiants textuels. Chaque critère est associé à une information booléenne indiquant s'il doit être maximisé ou minimisé et éventuellement à des valeurs de seuils p_j, q_j . Les seuils manquants sont considérés comme valant zéro. Les poids représentent les valeurs $w_j, \forall j \in \mathcal{J}$. Les performances sont définies par des valeurs numériques $g_j(a), \forall a \in A, j \in \mathcal{J}$. Il faut $p_j \geq q_j \geq 0$, sinon le composant renvoie une erreur. Le composant renvoie l'ensemble de paires d'alternatives $\{(a, b) \in A \times A\}$ accompagnées chacune de la valeur demandée, $\Psi(a, b)$, binaire ou entre 0 et 1 en fonction des seuils donnés.

Pour rappel, les valeurs calculées sont définies comme suit. Notons que si un critère est à minimiser, les valeurs de $g_j(b)$ et $g_j(a)$ doivent être inversées.

$$\Psi(a, b) = \sum_{j \in \mathcal{J}} w_j K_j(a, b),$$

où

$$K_j(a, b) = \begin{cases} 1 & \Leftrightarrow g_j(b) - g_j(a) \leq q_j, \\ 1 - \frac{(g_j(b) - g_j(a)) - q_j}{p_j - q_j} & \Leftrightarrow q_j < g_j(b) - g_j(a) \leq p_j, \\ 0 & \Leftrightarrow p_j < g_j(b) - g_j(a). \end{cases}$$

Le résultat représente donc la relation de concordance utilisée par ÉLECTRE III et ÉLECTRE TRI. Notons que le modèle ÉLECTRE IS (que nous n'avons pas présenté) utilise la même relation de concordance. La relation de concordance utilisée par ÉLECTRE IV et ÉLECTRE II s'obtient en choisissant comme valeurs de seuils $p_j = q_j = 0, \forall j \in \mathcal{J}$.

6.3.2 Préférence

Le composant *PrometheePreference* calcule une relation de préférence telle que définie par PROMÉTHÉE. Il fonctionne de manière analogue au composant *ElectreConcordance* : il accepte les mêmes entrées que ce composant, et calcule la valeur la fonction de préférence de PROMÉTHÉE de type P^5 associée à chaque paire d'alternatives. Les valeurs correspondant aux formes plus simples de fonctions de préférence P^1, P^2, P^3 s'obtiennent en sélectionnant les valeurs des seuils de manière appropriée. La famille de méthodes PROMÉTHÉE est décrite à l'annexe B.

Le fait que $P(a, b) = 1 - \Psi(b, a)$ est utilisé dans certains tests internes à la bibliothèque pour vérifier certains calculs.

6.3.3 Discordance

Le composant *ElectreDiscordances* calcule l'ensemble de relations de discordance D_j utilisées par ÉLECTRE III et ÉLECTRE TRI. Il prend en entrée un ensemble d'alternatives \mathbb{A} , un ensemble de critères \mathcal{J} muni d'informations concernant le sens de préférence, les performances $g_j(a), \forall a \in \mathbb{A}, j \in \mathcal{J}$, et des seuils de préférence et de véto p_j, v_j . Une valeur manquante pour p_j est interprétée comme $p_j = 0$. Une valeur de véto non spécifiée entraîne $D_j(a, b) = 0, \forall a, b \in \mathbb{A}$.

Les relations utilisées par ÉLECTRE IV et ÉLECTRE II peuvent être obtenues en choisissant $p_j = v_j$. Notons que rien n'oblige l'utilisateur à envoyer des mêmes valeurs de seuils aux composants *ElectreConcordance* et *ElectreDiscordances*. Il peut donc choisir $q_j = p_j = 0$ pour le premier et $p_j = v_j > 0$ pour le second.

La publication sur laquelle cet exposé est basé (Cailloux, 2010) définit la discordance différemment : dans le cas des modèles ÉLECTRE IV et ÉLECTRE II, $D_j^{\text{binaire}}(a, b) = 1 \Leftrightarrow g_j(b) - g_j(a) \geq v_j$. Pour autant qu'on postule $v_j > p_j$, hypothèse effectuée dans la publication en question, c'est une définition tout aussi raisonnable. Cependant, elle n'est pas conforme à la présentation originale de la méthode, ce qui risque d'introduire des confusions à l'usage. De plus cette définition interdit $v_j = p_j = q_j$, car on pourrait alors avoir $K_j(a, b) = 1$ et $D_j^{\text{binaire}}(a, b) = 1$, et requiert donc de considérer séparément des cas inclus dans cette présentation-ci comme cas particuliers des formules générales.

6.3.4 Surclassement

Le composant *ElectreOutranking* fournit une relation de surclassement telle que définie par (surclassement) page 45. Le composant demande en en-

6 Contributions sous forme de logiciels

trée une relation de concordance et une relation de discordance par critère. Si ces relations sont celles calculées par les composants *ElectreConcordance* et *ElectreDiscordances*, la relation calculée par *ElectreOutranking* est donc bien la relation S utilisée par ÉLECTRE III et ÉLECTRE TRI.

Un ensemble d'alternatives et de critères peut également être précisé en entrée du composant, mais cette information est facultative car elle peut être déduite des relations données en entrée.

Le résultat de ce calcul peut être envoyé à un composant *cutRelation* qui coupe la relation au dessus d'un certain seuil. En lui donnant la valeur du seuil de coupe λ , on obtient la relation binaire S^λ utilisée par ÉLECTRE TRI. En passant à ce composant les relations de concordance et de discordances calculées avec les paramètres adéquats indiqués ci-dessus, le résultat correspondra à la relation utilisée par ÉLECTRE IV.

6.3.5 Tri

Il est possible de calculer les affectations d'un modèle ÉLECTRE TRI à l'aide du composant *ElectreTriExploitation*. Le composant attend en entrée un ensemble d'alternatives, de critères, de profils et de catégories, de poids. Les performances des alternatives et des profils doivent également être fournies.

Il faut indiquer au composant si l'exploitation souhaitée est l'affectation pessimiste, l'affectation optimiste, ou les deux. Dans les deux premiers cas, le composant renvoie une affectation précise, donc une association de chaque alternative à exactement une catégorie. Le dernier cas fournit une association de chaque alternative à un intervalle de catégories.

Un dernier paramètre, facultatif, permet de demander de traiter la discordance de façon binaire. Cela permet de reproduire les affectations d'un modèle MR Sort. En l'absence du paramètre, la discordance est non binaire.

6.3.6 Flux

Les flux définis par la famille de modèles PROMÉTHÉE sont calculés par *PrometheeFlows*. Le composant peut calculer le flux positif, négatif ou net. Il demande un ensemble d'alternatives et de critères avec les seuils et poids liés, les performances des alternatives, et éventuellement un seuil de majorité, et une information indiquant le type de flux demandé. Il renvoie un ensemble d'alternatives associées chacune à une valeur.

6.4 Désagrégation d'un modèle MR Sort

Un service web implémentant le programme ICLV a également été publié. Il prend en entrée un ensemble d'alternatives, de critères, de profils et de catégories, les exemples des décideurs. Les profils doivent simplement être nommés, les évaluations ne doivent pas être fournies. Les performances des alternatives utilisées en exemple doivent être données en entrée. L'utilisateur peut autoriser ou interdire l'utilisation de véto dans le modèle calculé. Dans le second cas, le modèle sera de type MR Sort sans véto. L'utilisateur peut donner des seuils de véto en entrée et des valeurs de performances pour les profils, auquel cas ces valeurs sont utilisées pour contraindre le modèle résultant.

Le service web répond en indiquant si l'ensemble de contraintes spécifié peut être représenté dans un modèle de type MR Sort, avec ou sans véto selon la demande de l'utilisateur. Si la réponse est positive, le service renvoie également des valeurs pour les profils, et éventuellement les seuils de véto, partagées par l'ensemble des décideurs, et indique également un ensemble de valeurs de poids et de seuil de majorité pour chaque décideur.

Le composant admet, facultativement, un paramètre \underline{w} servant de valeur minimale pour les poids : si l'utilisateur fait usage de cette fonction, le programme cherchera un modèle en ajoutant les contraintes $w_j^t \geq \underline{w}, \forall j \in \mathcal{J}, t \in \mathcal{T}$. Dans l'exemple de la section suivante, les modèles cherchés ont été contraints avec $\underline{w} = 0,05$. Ce paramètre peut être utilisé pour tenter d'exclure les modèles de préférence qui accorderaient trop peu d'importance à certains poids (par exemple dans un cas où tous les décideurs pensent que tous les critères sont

Idéalement, dans un tel cas, les contraintes devraient assurer que tous les critères soient nécessaires à au moins une coalition, donc que tous les critères soient opérants (voir (critère opérant) page 231). L'utilisation d'un poids minimum n'est pas pleinement satisfaisante pour assurer cette condition : si le poids minimum est faible, la contrainte additionnelle n'assure pas que tous les critères soient opérants (cela se produit dans notre exemple où $\underline{w} = 0,05$), et si le poids minimum est grand, l'espace des modèles compatibles risque d'être réduit bien plus que nécessaire. Une piste de recherche future est de trouver comment assurer cette condition sans contraindre exagérément l'espace des modèles acceptables, et sans trop compliquer le programme à résoudre.

6.5 Convergence de modèles pour l'évaluation de risques

Cette section présente une application hypothétique de la méthode proposée à la section 3.1. L'application vise à construire une échelle d'évaluation de

6 Contributions sous forme de logiciels

risque. L'objectif est, d'une part, de suggérer que cette méthode peut être appliquée dans un domaine où l'AMCD est peu présente, de l'autre de montrer comment certains outils intégrés à la plateforme Decision Deck peuvent être utilisés pour la mettre en pratique. Cette section s'inspire de deux articles de conférence (Cailloux et Mousseau, 2011 ; Mayag *et al.*, 2011) et une version étendue de cette section a été soumise à Reliability Engineering & System Safety (RESS) (Cailloux *et al.*, 2012a).

Nous nous intéressons ici au problème de l'évaluation des risques associés à des zones de territoires, prenant en compte le point de vue de plusieurs décideurs. L'analyse de risques intéresse les chercheurs depuis bien longtemps et de nombreuses méthodologies ont été proposées pour aborder ce domaine. Le concept de probabilité joue souvent un rôle important pour représenter, analyser et évaluer le risque associé à des événements aléatoires, mais une tendance récente vise à réduire le recours aux probabilités. Aven (2010, page 31) par exemple fait remarquer que la détermination de probabilités s'appuie sur un certain nombre d'hypothèses et de suppositions qui cachent souvent des incertitudes. Une fois les probabilités déterminées, l'attention risque de se focaliser sur celles-ci et non plus sur les incertitudes sous-jacentes.

Aven (2003, 2008) offre une revue détaillée de démarches d'analyse de risques. La littérature propose des méthodes basées sur des réseaux bayésiens, sur la décision dans l'incertain ou dans le risque, sur les arbres de défaillance, etc.

Notre perspective propose une évaluation qualitative de la notion de risque, ce qui n'implique pas nécessairement des résultats moins précis qu'une analyse quantitative pour les raisons indiquées ci-dessus. En outre, elle permet la prise en compte de la subjectivité des décideurs.

Cette section propose de représenter le problème de la construction d'une échelle de risque comme un problème d'obtention d'une fonction de tri multi-critère. Les alternatives correspondent aux zones territoriales à évaluer. Les différents niveaux de risques envisagés correspondent aux catégories, par exemple {Risque élevé, Risque moyen, Risque faible}. Ces niveaux devraient être associés à des mesures de précaution précises, de manière à donner aux catégories une sémantique précise. Les zones seront donc soumises à des mesures de précaution en fonction de leur niveau de risque. Les critères du problème de tri sont les points de vue du problème d'évaluation des risques, par exemple la présence d'une école dans la zone ou la proportion de personnes vulnérables. La section suivante décrit un processus d'aide à la construction d'une échelle de risque, et la section finale montre une façon d'utiliser les outils logiciels disponibles pour implémenter l'aide.

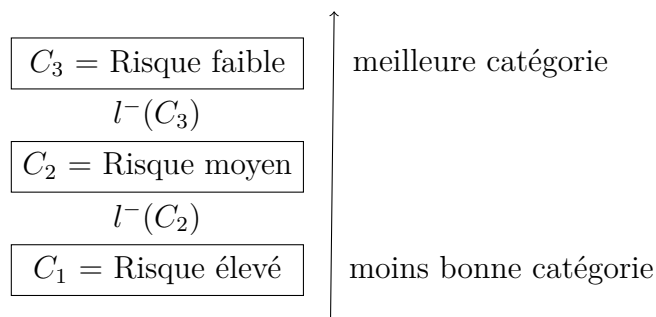


FIGURE 6.2 – Les trois catégories de risque et les profils séparateurs.

6.5.1 Un scénario de construction d'une échelle de risque

Un groupe de quatre décideurs souhaite développer une échelle pour évaluer les niveaux de risques associés à un danger particulier concernant une installation industrielle, pour chaque zone géographique à proximité de l'installation. Chaque zone doit être associée à une des trois catégories {Risque élevé \triangleleft Risque moyen \triangleleft Risque faible}, chaque catégorie étant associée à des mesures de précaution particulières. Les catégories sont également notées $C_1 \triangleleft C_2 \triangleleft C_3$ pour simplifier les notations. La figure 6.2 indique la direction de préférence des catégories et les deux profils séparateurs. Disposer d'une telle échelle de risques est utile par exemple lors de l'élaboration d'un Plan de Prévention des Risques Technologiques utilisé en France (<http://www.developpement-durable.gouv.fr/Maitrise-de-l-urbanisation-PPRT,12775.html>).

Les quatre membres du groupe considèrent que les six critères suivants devraient être utilisés pour évaluer le risque associé à chaque zone. Chaque critère est défini de telle sorte qu'une valeur plus élevée soit meilleure, donc indique un risque plus faible.

- pr La proportion de la population non vulnérable, en pourcents. Les enfants de moins de quinze ans et les personnes âgées sont considérées comme vulnérables.
- sc Une évaluation binaire indiquant l'absence d'école dans la zone. Un 0 indique la présence d'une école, 1 indique l'absence d'école.
- i Impact sur les autres installations industrielles, ce qui pourrait conduire à des effets en cascade. Évalué sur une échelle ordinale à cinq points.
- bv Vulnérabilité des bâtiments, évaluée sur une échelle ordinale à cinq points.
- pe Quantité de biens destructibles particulièrement précieux dans la zone, évaluée sur une échelle ordinale à cinq points.
- d Distance à l'installation industrielle, évaluée sur une échelle ordinale à

6 Contributions sous forme de logiciels

dm	Zone	pr	sc	i	bv	pe	d	Niveau de risque
dm1	Zone 08	91	1	4	5	4	1	Faible
dm1	Zone 11	93	0	3	5	3	2	Moyen
dm1	Zone 12	43	1	2	1	3	3	Élevé
dm1	Zone 13	91	0	5	2	4	3	Élevé
dm2	Zone 00	64	1	4	3	3	2	Moyen
dm2	Zone 01	84	0	3	2	5	3	Élevé
dm2	Zone 03	4	1	3	2	1	2	Élevé
dm2	Zone 05	14	1	4	5	5	3	Faible
dm3	Zone 00	64	1	4	3	3	2	Moyen
dm3	Zone 01	84	0	3	2	5	3	Moyen
dm3	Zone 06	9	1	2	2	2	1	Élevé
dm3	Zone 08	91	1	4	5	4	1	Faible
dm4	Zone 02	69	1	5	3	1	2	Moyen
dm4	Zone 05	14	1	4	5	5	3	Élevé
dm4	Zone 09	38	1	3	5	4	3	Élevé
dm4	Zone 10	36	0	3	2	4	1	Élevé

TABLE 6.1 – Une partie des exemples d’affectation initiaux.

Profil	pr	sc	i	bv	pe	d
$l^-(C_3)$	84	1	4	4	4	3
$l^-(C_2)$	41	1	3	3	2	2

TABLE 6.2 – Profils inférés. Le premier profil sépare les catégories « Risque faible » et « Risque moyen » et le second sépare les catégories « Risque moyen » et « Risque élevé ».

trois points.

Nous supposons que chaque décideur fournit trente exemples de zones et leurs évaluations de niveau de risque associées. Ces exemples pourraient correspondre à de vraies zones que les décideurs ont précédemment évalués, ou à des zones fictives définies par leur vecteur d’évaluation. Une partie de ces exemples est reprise dans la table 6.1. L’entièreté des données utilisées dans cet exemple sont jointes au flux de composants publié sur <http://www.diviz.org/workflow.ressArticle.html>.

Le PM ICL décrit en section 3.1.3 est utilisé pour trouver des profils partagés par les décideurs satisfaisant leurs exemples d’affectation. Les résultats sont affichés dans la table 6.2.

La table 6.3 montre, pour chaque décideur, un ensemble de poids satisfaisant

6.5 Convergence de modèles pour l'évaluation de risques

DM	pr	sc	i	bv	pe	d	λ
dm1	0,05	0,204	0,05	0,348	0,254	0,094	0,649
dm2	0,05	0,273	0,05	0,223	0,13	0,273	0,774
dm3	0,237	0,237	0,237	0,144	0,05	0,094	0,572
dm4	0,556	0,05	0,05	0,244	0,05	0,05	0,804

TABLE 6.3 – Un ensemble de poids, trouvé par ICL, satisfaisant avec les valeurs de profils indiqués les exemples de tous les décideurs.

Profil	pr	sc	i	bv	pe	d
$l^-(C_3)$	82	1	5	4	4	3
$l^-(C_2)$	41	1	3	3	2	2
$\nu^{\text{BM}}(C_3)$	-	-	1	-	-	-
$\nu^{\text{BM}}(C_2)$	-	-	-	-	-	-

TABLE 6.4 – Profils inférés lors de la deuxième itération.

les exemples d'affectation avec les profils trouvés. Notons que ce ne sont pas les seules valeurs de poids possibles qui reproduisent les exemples.

Cependant, le groupe n'est pas pleinement satisfait par ces résultats. Il trouve que le critère *i* ne devrait se prononcer en faveur de l'affectation d'une alternative au niveau de risque le plus faible que lorsque la performance de l'alternative est maximale sur ce critère. Ceci peut être pris en compte en contraignant la valeur de performance du profil correspondant. De plus, le groupe s'accorde à penser qu'une performance de seulement 1 sur ce même critère devrait interdire l'accès de cette zone à la meilleure catégorie (« Risque faible »), quelles que soient les performances sur les autres critères. Ceci peut être modélisé à l'aide d'un veto. Le PM ICLV est alors utilisé pour chercher un modèle de préférence satisfaisant tous les exemples et ayant des valeurs $\nu_1^{\text{BM}}(C_3) = 1$ et $l_1^-(C_3) = 5$. Les nouvelles valeurs de profils et l'ensemble de poids correspondants sont indiqués par les tables 6.4 et 6.5.

En supposant que le groupe accepte les valeurs profils trouvées, l'approche de

DM	pr	sc	i	bv	pe	d	λ
dm1	0,05	0,221	0,05	0,365	0,221	0,094	0,633
dm2	0,05	0,512	0,05	0,144	0,05	0,194	0,854
dm3	0,237	0,237	0,237	0,144	0,05	0,094	0,572
dm4	0,244	0,05	0,05	0,556	0,05	0,05	0,804

TABLE 6.5 – Poids correspondants à la deuxième itération.

6 Contributions sous forme de logiciels

DM	pr	sc	i	bv	pe	d	λ
tous	0,36	0,18	0,18	0,18	0,05	0,05	0,68

TABLE 6.6 – Un ensemble de poids satisfaisant, avec les profils inférés lors de la deuxième itération, 96 exemples parmi les 120 de départ.

Zone	pr	sc	i	bv	pe	d	Niveau de risque
Zone A	91	1	4	5	4	1	Moyen
Zone B	93	0	3	5	3	2	Élevé
Zone C	43	1	2	1	3	3	Faible
Zone D	91	0	5	2	4	3	Élevé
Zone E	64	1	4	3	3	2	Élevé
Zone F	84	0	3	2	5	3	Élevé

TABLE 6.7 – Évaluations des niveaux de risques relatifs à six zones.

Damart *et al.* (2007) peut ensuite être utilisée pour construire progressivement des valeurs de poids consensuelles parmi le groupe de décideurs. Nous supposons ici que le résultat de cette procédure génère les poids affichés en Table 6.6, qui, utilisés avec les profils et vétos affichés en Table 6.4, reconstituent 96 exemples d'affectation des décideurs sur les 120 de départ.

À ce stade, le groupe s'est mis d'accord sur une fonction de tri qui peut être utilisée pour évaluer les risques de nouvelles zones. Par exemple, la table 6.7 indique les affectations de six zones résultant de l'utilisation du modèle obtenu.

6.5.2 Implémentation du scénario

Montrons maintenant comment les résultats peuvent être obtenus à l'aide des services web XMCDAs accessibles depuis diviz. La figure 6.3 présente l'enchaînement de composants utilisé pour le processus d'élicitation et pour l'affectation des zones A à F avec le modèle élicité. Les grandes boîtes aux coins arrondis représentent les composants logiciels, les petits rectangles représentent les fichiers de données au format XMCDAs. Le flux contient deux instances du composant *ElectreTri1GroupDisaggregationSharedProfiles* (1 et 2) représentant l'élicitation des profils partagés lors des deux itérations du scénario. La sortie de la première instance n'est pas réutilisée, car les décideurs n'en étaient pas satisfaits. Le composant *criteriaDescriptiveStatistics* (3) donne aux décideurs et à l'analyste des informations concernant les alternatives utilisées dans les exemples d'affectation : sont affichés pour chaque critère, la valeur moyenne, l'écart type, la performance minimale et maximale.

6.5 Convergence de modèles pour l'évaluation de risques

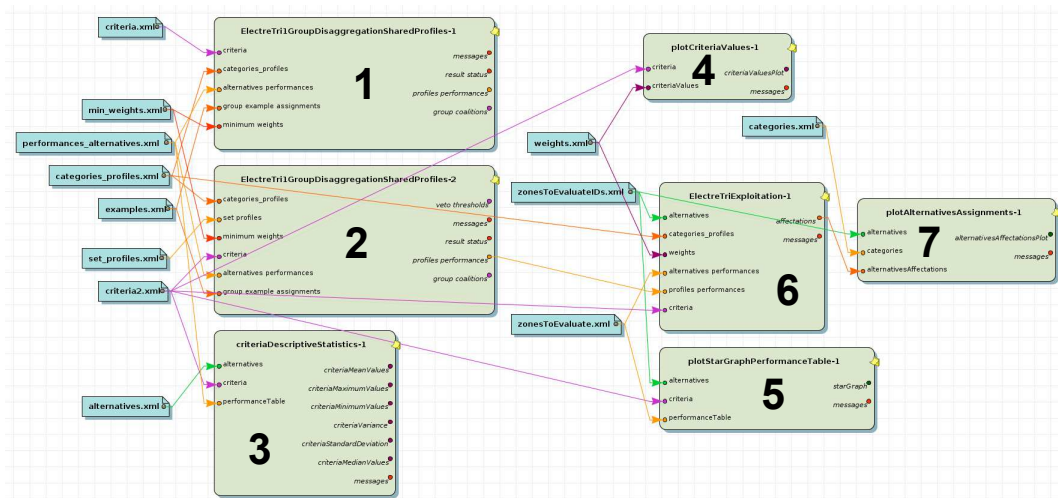


FIGURE 6.3 – L'enchaînement de composants utilisé dans diviz pour implémenter le scénario proposé.

Une fois les résultats du second composant (2) validés, la procédure de Darnat *et al.* (2007) est utilisée pour obtenir un consensus sur les poids. Cette procédure n'est pas encore disponible dans diviz. Les poids résultants sont affichés par le composant *plotCriteriaValues* (4), repris dans la partie gauche de la figure 6.4.

Les six zones de la table 6.7 sont représentées graphiquement par le composant *plotStarGraphPerformanceTable* (5). L'affichage correspondant à la zone C (resp. B) est représenté au milieu (resp. à droite) de la figure 6.4. Le résultat du composant d'élicitation des profils, ainsi que les

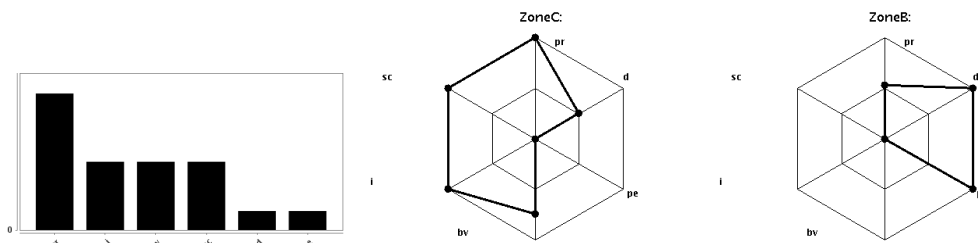


FIGURE 6.4 – Poids consensuels des critères et graphes en étoile de la zone C (resp. B) affectée à la catégorie « Faible » (resp. « Élevé ») par le modèle de préférence inféré.

6 Contributions sous forme de logiciels

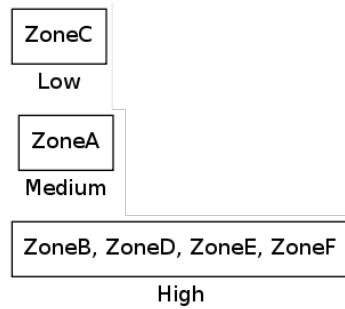


FIGURE 6.5 – Les affectations des zones A à F.

poils partagés, sont ensuite envoyés au composant *ElectreTriExploitation* (6), qui associe ces six zones à leur niveau de risque. Enfin, le composant *plotAlternativesAssignment* (7) affiche ces affectations ainsi que représenté par la figure 6.5.

Le flux présenté ici peut être téléchargé depuis le site de diviz à l'adresse <http://www.diviz.org/workflow.ressArticle.html>, voir également l'annexe D de ce document, numéro 5. Le lecteur intéressé peut ensuite importer ce flux à l'aide de son installation locale de diviz pour reproduire les calculs proposés dans cet exemple. Des variations peuvent cependant être observées dans les résultats en fonction de la solution retenue par les PMs ICL et ICLV : ces programmes déterminent un ensemble de solutions admissibles mais la sélection d'une solution parmi cet ensemble n'est pas déterministe.

7 Modèle de données pour l'AMCD

Un aspect qui nous tient à cœur concernant notre collaboration à l'entreprise de Decision Deck est que nous avons proposé une amélioration du schéma XMCDA, et proposé au consortium de le publier comme une version beta d'une nouvelle version (version 3) du schéma. Au moment où ces lignes sont écrites, le consortium examine la proposition. Le contenu de ce chapitre a été soumis à DSS (Cailloux *et al.*, 2012c).

7.1 Introduction

Nous proposons dans ce chapitre un nouveau modèle de données pour la représentation de données intervenant dans des méthodes d'AMCD, et en proposons une implémentation en XML. Il vise à améliorer certains aspects de la version actuelle du schéma XMCDA (version 2). Mentionnons que l'existence d'un modèle de données XMCDA n'est pas seulement utile à l'échange de données entre services web XMCDA tels qu'hébergés par le consortium Decision Deck. Plus généralement, un tel modèle de données, s'il est suffisamment robuste, peut également servir à l'échange de données entre des logiciels complets implémentant des méthodes d'aide à la décision, ou à l'intégration par des systèmes d'entreprise (utilisant éventuellement uniquement des composants développés en interne) de données de préférence.

Notre proposition vise principalement deux objectifs d'amélioration par rapport au schéma XMCDA actuellement utilisé.

7.1.1 Différents schémas pour différents usages

Premièrement, la proposition s'écarte de ce qui nous semble être une grosse faiblesse de l'approche actuelle, qui utilise un seul schéma pour tout le monde. De ce fait, les composants ne peuvent pas être fortement typés : ils ne précisent pas quel type de données exactement ils attendent ou ils produisent. Le schéma annonce simplement qu'ils communiquent en XMCDA, sans qu'il soit précisé s'ils acceptent un tableau de performance, une fonction de valeur, des valeurs de paramètres de seuils, un ensemble d'alternatives, etc., toutes choses qui

peuvent être représentées dans le schéma XMCDA. Un problème concret illustrant cet état de fait est que l'utilisatrice doit indiquer au programme *diviz*, lors de la construction de l'enchaînement des composants, quelle sortie se connecte à quelle entrée. Ceci est nécessaire car ces informations ne peuvent être déduites des données fournies dans le schéma. Plus généralement, un typage fort fournirait d'autres fonctionnalités intéressantes. Le programme de construction d'enchaînement de composants pourrait prévenir un grand nombre d'erreurs en constatant l'inadéquation entre deux types que l'utilisateur essaie de connecter. Cela simplifierait le travail du développeur de composants, qui ne devrait pas gérer ces erreurs lui-même. Un autre exemple est qu'un logiciel pourrait guider l'utilisatrice dans le choix des services web XMCDA disponibles en mettant en avant les composants qui requièrent uniquement des données disponibles dans le logiciel.

Pour être complet, il faut préciser que *diviz* a connaissance de méta-données qui contiennent l'information de typage fort cherchée, mais elles ne sont actuellement pas exploitées pour fournir les fonctionnalités décrites, et il semble raisonnable d'utiliser des mécanismes classiques prévus par le langage XML plutôt que de re-programmer un mécanisme de typage « maison ».

7.1.2 Un encodage moins ambigu

Un deuxième objectif de notre proposition est de réduire l'ambiguïté présente dans le schéma XMCDA. Par choix de conception, XMCDA est un schéma très vaste : il doit être capable de représenter toute donnée dont un développeur pourrait avoir besoin dans une méthode d'aide multicritère à la décision. Cela en fait un schéma très général, dont un inconvénient, à notre avis inévitable suite au choix de conception large qui a été pris, est l'introduction de multiples façon d'encoder un même concept. Notre approche propose plutôt de travailler en deux niveaux. Le schéma central est plus restreint (bien que déjà d'une certaine généralité) et se concentre sur la représentation de concepts importants en AMCD. Le développeur peut utiliser les types disponibles dans ce schéma central, mais également en proposer d'autres, spécifiques à la méthode qu'il développe ou à une famille de méthodes peu connue, qui sont peut-être seulement utilisés par quelques développeurs dans son laboratoire. Il peut dans ce cas proposer un schéma annexe, soit en le publiant (le consortium Decision Deck pourrait par exemple publier les schémas les plus populaires), soit en le partageant simplement avec ses collaborateurs.

L'inconvénient de cette approche est que certains développeurs pourraient oublier de vérifier si un type a déjà été développé dans un schéma annexe, et re-développer le leur, entraînant donc une rupture de compatibilité avec d'autres composants. Mais cela n'arrivera que pour des types de moindre importance,

et nous pensons que ce risque est également présent à cause des ambiguïtés présentes dans la version 2 de XMCD (qui présente également ce risque pour des types majeurs). De plus, il est vraisemblable qu'une limite devra être imposée tôt ou tard à l'extension du schéma XMCD, même si l'approche actuelle de représentation large est maintenue.

7.1.3 Réutilisation des mécanismes XML classiques

Outre ces deux points qui nous semblent constituer les principales améliorations de l'approche existante, notre proposition utilise, et permet aux développeurs qui l'adopteraient d'utiliser, des concepts avancés du langage XML tels que les requêtes xpath (ceci sans parler de l'utilisation des namespace qui implémente l'objectif décrit ci-dessus). Cependant, un objectif important de conception du schéma XMCD que nous souhaitons conserver est qu'il doit être simple d'utilisation, car la communauté de programmeurs de services web XMCD est principalement composée de chercheurs en AMCD, souvent mathématiciens et non développeurs professionnels. Les fonctionnalités avancées telles que l'utilisation de xpath sont disponibles pour les développeurs experts qui souhaitent en faire usage (par exemple dans une entreprise), et peuvent être laissées de côté par d'autres développeurs sans que cela entraîne d'inconvénients par rapport à l'utilisation de la version actuelle de XMCD.

7.1.4 Portée du modèle de données et nomenclature

Le modèle de données proposé permet de représenter les concepts nécessaires à MAVT, aux méthodes ÉLECTRE III, ÉLECTRE TRI et à leurs extensions stochastiques SMAA-2/TRI (Lahdelma *et al.*, 1998; Lahdelma et Salminen, 2001; Tervonen *et al.*, 2009), et à AHP. Ce périmètre couvre les deux structures de préférence appliquées dans la majorité des applications (Wallenius *et al.*, 2008), la théorie de la valeur et le surclassement, et les trois problématiques habituellement considérées : choix, rangement, et tri, et donc la plupart des méthodes utilisées en pratique (Wallenius *et al.*, 2008).

Notons que le modèle de données adopte le terme utilité plutôt que valeur pour désigner les critères associés à des fonctions de valeurs. Cela évite le risque de confusion avec d'autres types utilisant le terme de valeur, et permet de souligner que le modèle s'applique également à la théorie de l'utilité (au sens de la décision dans le risque). Bien que la théorie de l'utilité et la théorie de la valeur désignent deux concepts distincts, il n'y a pas lieu de dupliquer les entités du point de vue de l'encodage des données nécessaires à l'application de ces modèles.

Le modèle de données fait référence au concept d'attribut, à distinguer du critère, dans un sens légèrement différent de celui indiqué à la section 1.2.4. Pour le modèle de données, un ensemble de performances (représentant les valeurs de la fonction g_j dans le cas où elle s'applique à un ensemble fini d'alternatives) est toujours associé à un attribut, même lorsque l'objet que les valeurs de performances décrivent est conceptuellement associé à une préférence. Il n'est en effet pas nécessaire de distinguer dans le modèle de données le concept d'attribut et de critère lorsque les données à encoder ne varient pas, comme dans le cas d'un ensemble de performances. Par contre, le critère est utilisé dans le modèle de données lorsque des données de préférence doivent être encodées. Cette séparation permet de plus simplement dupliquer les informations de préférence, par exemple pour représenter les préférences de plusieurs décideurs, sans pour autant dupliquer les données liées à l'attribut. Ainsi, pour représenter un critère associé à des performances et à des informations de préférences, il faudra utiliser une entité attribut du modèle de données ainsi qu'une entité critère.

Les diagrammes de ce chapitre sont conformes à la notation UML 2.0 : les entités (Section 7.2) sont représentées par des diagrammes de classe, les instantiations (Section 7.3) comme des diagrammes objets, et les flux (Section 7.3) comme des diagrammes de composants. Les héritages sont implémentés en XML via le mécanisme d'extension.

La section 7.2 détaille le modèle de données. La section suivante donne des exemples d'utilisation précis du modèle de données, et la section 7.4 présente une discussion concernant certains choix effectués et les modalités générales possibles d'usage du modèle de données.

7.2 Modèle de données

7.2.1 Entités à clés

Les entités les plus importantes du modèle sont les quatre **KeyedEntity** : **alternative**, **attribute**, **criterion** et **category** indiquées sur la figure 7.1. Une telle entité est munie d'un identifiant qui lui sert de nom et d'identifiant unique permettant de s'y référer. L'héritage nous permet d'indiquer que ces quatre entités peuvent être utilisées partout où nous faisons référence à une entité à clé.

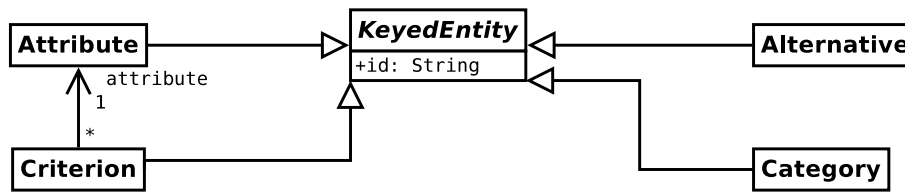


FIGURE 7.1 – Les entités à clé.

7.2.2 Mesures

Le modèle de données définit différents types de mesures, présentées dans la figure 7.2, utilisées pour encoder entre autres les performances des critères. L'héritage nous permet d'assurer la substituabilité des différentes mesures et permet l'entrée uniforme de différentes mesures précises ou imprécises dans un modèle de type Stochastic Multicriteria Acceptability Analysis (SMAA). Beaucoup de modèles d'AMCD utilisent certains paramètres imprécis ou incertains, et ceux-ci sont souvent représentés à l'aide de distributions de probabilités discrètes ou continues. Les informations incertaines discrètes peuvent être encodées comme des mesures nominales imprécises, et les informations incertaines continues à l'aide des intervalles ou des mesures gaussiennes. Notons que le modèle de données n'inclut pas d'autres distributions pour représenter des mesures imprécises, car cela étendrait le modèle très fortement. Rappelons notre objectif de proposer un modèle central simple et de permettre aux utilisateurs de définir des types additionnels dans des espaces de nommage externes grâce aux mécanismes d'extension classiques du XML.

7.2.3 Relations

Deux entités à clés peuvent être mises en relation avec une mesure grâce à une paire valuée, par exemple l'alternative « car1 » peut avoir une valeur 120,0 concernant l'attribut « vitesse », représentant sa vitesse maximale, ou les alternatives « car1 » et « car2 » peuvent être associées à la mesure binaire « vrai » pour représenter une affirmation de préférence holistique $\text{car1} \succ \text{car2}$.

Comme le mécanisme d'héritage XML n'admet pas de typage générique, le type **ValuedPair** est répété pour chaque type de mesure, ce qui permet de spécifier exactement à quel type de valeur la relation se réfère. Au long de ce chapitre, une entité associée à une mesure de type M est préfixée par la notation $\langle M \rangle$. Lorsque le préfixe est omis, la mesure est de type **ExactMeasurement**, et l'entité **NominalValuedPair** désigne le type **ValuedPair** associé à une mesure de type **NominalMeasurement**. Dans l'implémentation XML, le type est défini pour le type de mesure abstrait de base puis répété pour chaque type de

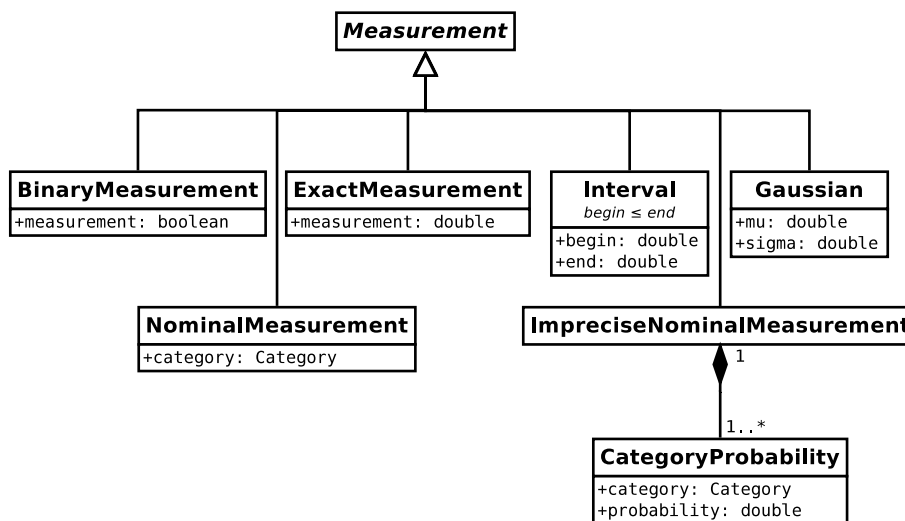


FIGURE 7.2 – Data model measurements.

mesure (exacte, intervalle, ...). La remarque vaut pour tous les types préfixés de la notation $\langle M \rangle$.

Une entité évaluée représente une mesure associée à une entité à clé seulement, par exemple un critère peut avoir un poids.

Des ensembles de paires évaluées forment des relations évaluées, qui représentent une relation entre deux ensembles d'objets, par exemple des performances d'un ensemble d'alternatives sur un ensemble d'attributs. Notons que de même que les relations évaluées qui représentent des ensembles de paires évaluées, le schéma XML comprend également des ensembles d'entités évaluées. Nous ne parlons pas en détail des types représentant simplement des ensembles homogènes d'autres types, qui seront présentés brièvement en section 7.2.6, sauf pour ce qui concerne les relations évaluées car elles ont une sémantique et une importance particulières en AMCD.

La figure 7.3 présente les paires, entités, et relation évaluées.

Associer de simples entités à clés à des mesures comme nous le faisons, plutôt que des types spécifiques comme **alternative**, entraîne une perte de sécurité de typage (type safety), puisque le schéma ne peut plus préciser quel type d'entité il utilise. Une solution pratique est de contraindre les références à des ensembles d'entités à clés contraints à être du bon type grâce à l'utilisation des mécanismes XML XPath et des clés XML. Le listing 7.1 montre un exemple d'un tel schéma définissant un ensemble d'alternatives, un ensemble d'attributs, un ensemble de paires évaluées, et des contraintes de chemins (path constraints) assurant que les paires évaluées font bien référence à une alternative et à un attribut. Sans ces contraintes le schéma serait également valide si

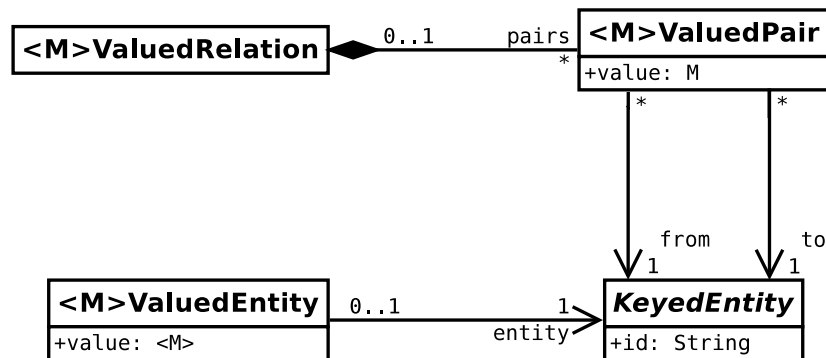


FIGURE 7.3 – Les entités, paires et relations valuées. Le préfixe $\langle M \rangle$. représente un type de mesure.

certaines paires faisaient référence, par exemple, à deux alternatives.

Lorsque la relation valuée utilise le type de mesure binaire, la relation acquiert une sémantique différente puisqu'elle peut alors représenter une relation binaire au sens mathématique. Afin d'augmenter la lisibilité et l'efficacité de l'encodage, la relation binaire définit la mesure comme facultative : une mesure manquante doit être considérée, par convention, comme ayant la valeur Vrai. De même, quand le contexte rend clair l'ensemble sur lequel la relation binaire est supposée définie, par convention, les paires manquantes doivent être considérées comme associées à la valeur Faux. Ces règles permettent de représenter une relation binaire plus naturellement car seules les paires qui sont dans la relation doivent être représentées, comme l'illustre le listing 7.2. Les relations binaires peuvent également être utilisées pour représenter des affectations binaires dans un problème de tri en utilisant la relation sur un ensemble d'alternatives et un ensemble de catégories. Les contraintes de chemin peuvent être utilisées pour assurer qu'une alternative est affectée à exactement une catégorie si des affectations précises sont requises.

7.2.4 Critères

Le modèle de données définit un critère comme une information de préférence liée aux mesures associées à un certain attribut. Par exemple, si l'attribut est la vitesse maximale, il nous faut un critère indiquant la direction de préférence : soit implicitement, via un critère associé à une fonction de valeur croissante, soit explicitement, avec un critère directionnel. Séparer le critère de l'attribut introduit un peu de répétition lorsque exactement un critère est associé à chaque attribut, et le fait de pouvoir associer un identifiant particulier à un critère, distinctement de l'identifiant de l'attribut, n'est pas utile dans ce cas.

7 Modèle de données pour l'AMCD

```
<element name="exampleOne">
  <complexType>
    <sequence>
      <element name="alternativeSet" type="x3:alternativeSetType" />
      <element name="attributeSet" type="x3:attributeSetType" />
      <element name="performanceTable" type="x3:valuedRelationType" />
    </sequence>
  </complexType>

  <key name="alternativeKey">
    <selector xpath="alternativeSet/alternative" />
    <field xpath="id" />
  </key>
  <key name="attributeKey">
    <selector xpath="attributeSet/attribute" />
    <field xpath="id" />
  </key>
  <keyref name="performanceTableAlternativeKeyRef" refer="x3e:alternativeKey">
    <selector xpath="performanceTable/valuedPair/from" />
    <field xpath="@ref" />
  </keyref>
  <keyref name="performanceTableAttributeKeyRef" refer="x3e:attributeKey">
    <selector xpath="performanceTable/valuedPair/to" />
    <field xpath="@ref" />
  </keyref>
</element>
```

Listing 7.1 – Un schéma XML avec des contraintes de chemin.

Mais cette séparation permet d'associer différents modèles de préférence à un même ensemble de performances, ce qui peut être utile entre autres dans un contexte de décision de groupe. Les différents types de critères sont présentés dans la figure 7.4.

Le modèle de données permet d'utiliser un critère défini sur des mesures cardinales avec des alternatives évaluées par des mesures nominales (des catégories). Pour ce faire, il est nécessaire de recoder les performances nominales en performances cardinales, soit d'associer à l'attribut nominal initial un nouvel attribut cardinal, et d'indiquer la relation entre les performances nominales et les nouvelles performances cardinales. Ceci est représenté dans le modèle de données à l'aide d'entités de type **ValuedEntity** associant les catégories aux mesures cardinales, et d'une entité **NominalToCardinal** associant cette entité à l'attribut nominal et au nouvel attribut. Cette entité est représentée en Figure 7.5. La section suivante illustre l'utilisation de ce type.

7.2.5 Fonctions

Le modèle de données contient trois formes classiques de fonctions permettant d'encoder les fonctions de valeurs partielles v_j . Les fonctions affines de la forme $v_j(x) = ax + b$ requièrent seulement les paramètres de pente et de

```

<exampleTwo>
  <alternativeSet>
    <alternative>
      <id>a1</id>
    </alternative>
    <alternative>
      <id>a2</id>
    </alternative>
    <alternative>
      <id>a3</id>
    </alternative>
  </alternativeSet>

  <relation>
    <valuedPair>
      <from ref="a1" />
      <to ref="a2" />
    </valuedPair>
    <valuedPair>
      <from ref="a2" />
      <to ref="a3" />
    </valuedPair>
    <valuedPair>
      <from ref="a3" />
      <to ref="a1" />
    </valuedPair>
    <valuedPair>
      <from ref="a3" />
      <to ref="a2" />
    </valuedPair>
  </relation>
</exampleTwo>

```

Listing 7.2 – Une instance XML (sans l'en-tête et les déclarations des espaces de nommage) contenant un ensemble d'alternatives et une relation binaire. En supposant que la relation représente une relation binaire irréflexive, on peut déduire que les paires $(a1, a3)$ et $(a2, a1)$ ne font pas partie de la relation.

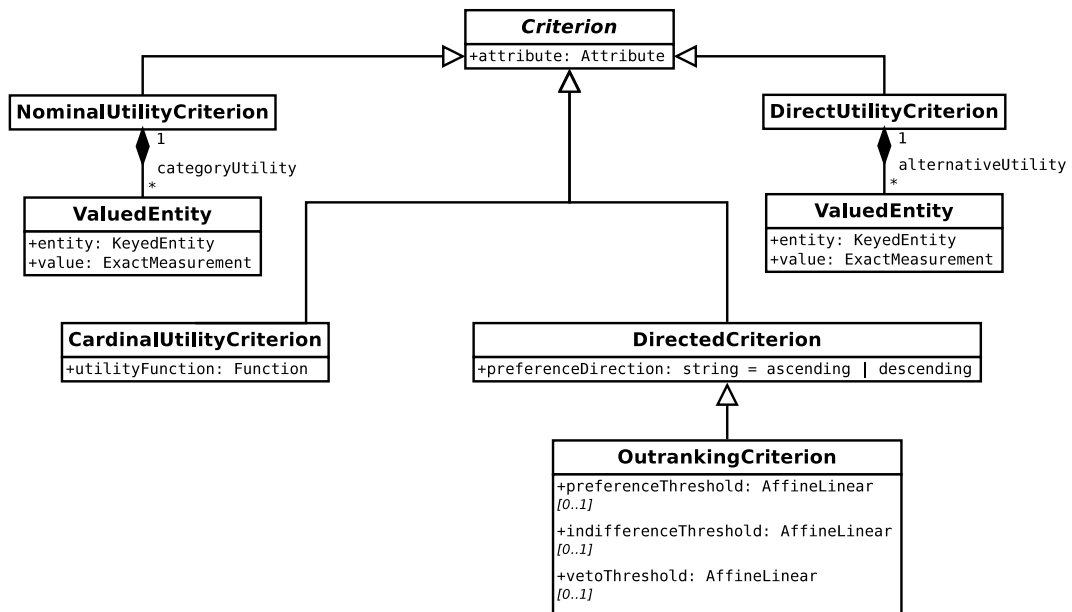


FIGURE 7.4 – Les critères compris dans le modèle de données.

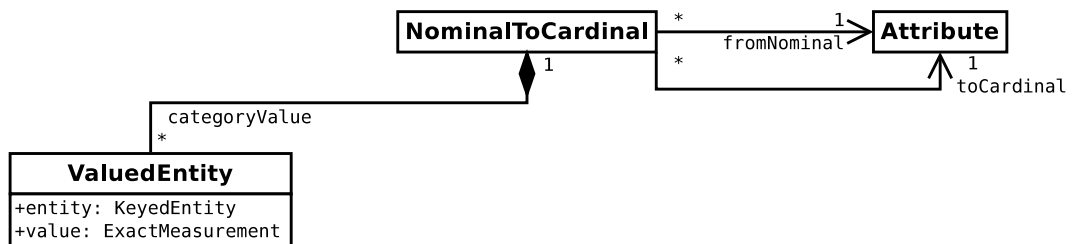


FIGURE 7.5 – Représentation des données de conversion d'un attribut nominal à un attribut cardinal.

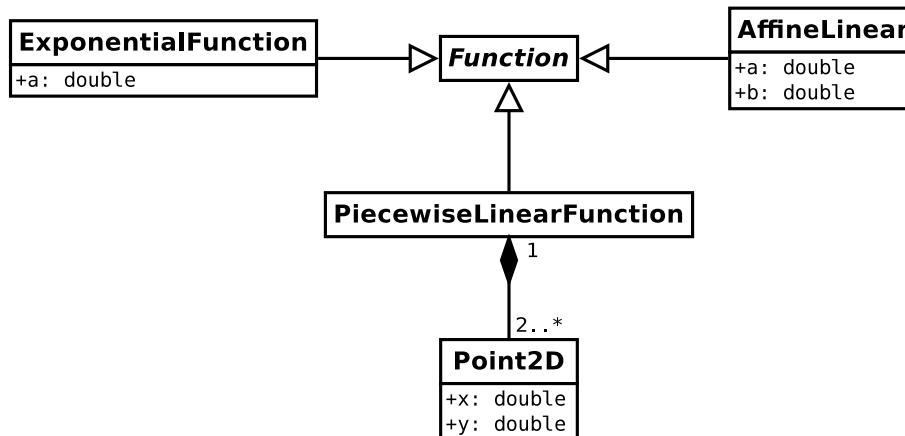


FIGURE 7.6 – Les fonctions.

décalage (ou ordonnée à l'origine). Une fonction linéaire par morceau peut être définie en utilisant au moins une paire de points à deux dimensions. La fonction de type exponentielle $v_j(x) = 1 - e^{-ax}$ requiert seulement un paramètre. La figure 7.6 montre ces trois types.

7.2.6 Entités représentant des ensembles d'autres entités

Le modèle de données contient des types supplémentaires représentant des ensembles homogènes d'autres types : **AlternativeSet**, **CategorySet**, **AttributeSet**, **CriterionSet** qui peut contenir tout autre type de critère, **Out-rankingCriterionSet** qui peut contenir seulement des critères de surclassement, les entités correspondantes aux autres types de critères étant similaires, et **ValuedEntitySet**.

7.2.7 Vue d'ensemble

La figure 7.7 présente une vue d'ensemble du modèle de données, sans la hiérarchie de mesures définie à la section 7.2.2 et sans les types d'ensembles indiqués à la section 7.2.6.

7.3 Utilisation du modèle de données

Cette section présente des cas d'utilisation du modèle de données. Nous illustrons d'abord avec quelques exemples, correspondant à des cas d'utilisation UML (*use cases*), l'encodage de mesures de performance et l'utilisation de types

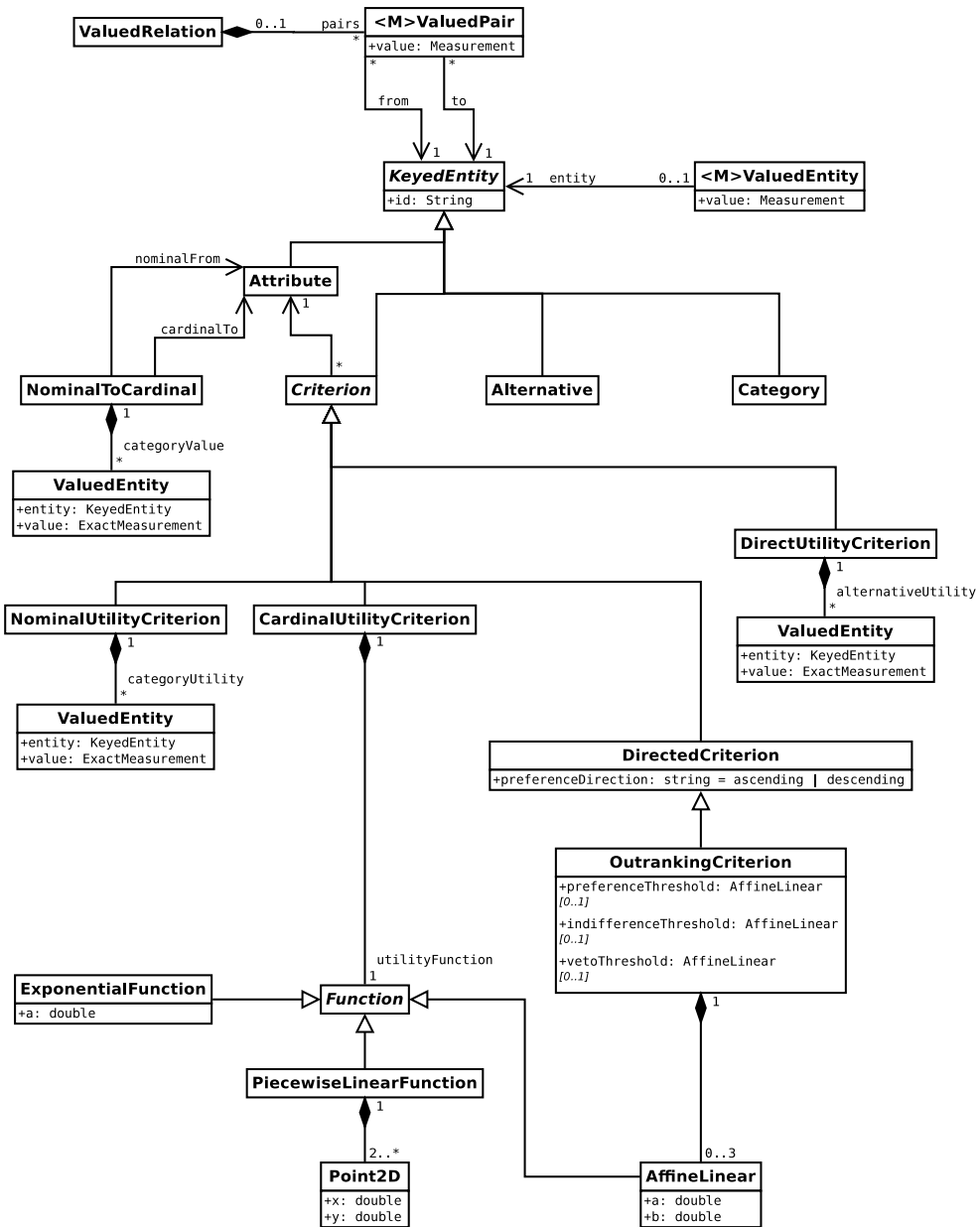


FIGURE 7.7 – Vue d'ensemble du modèle de données, sans la hiérarchie de mesures et sans les types d'ensembles.

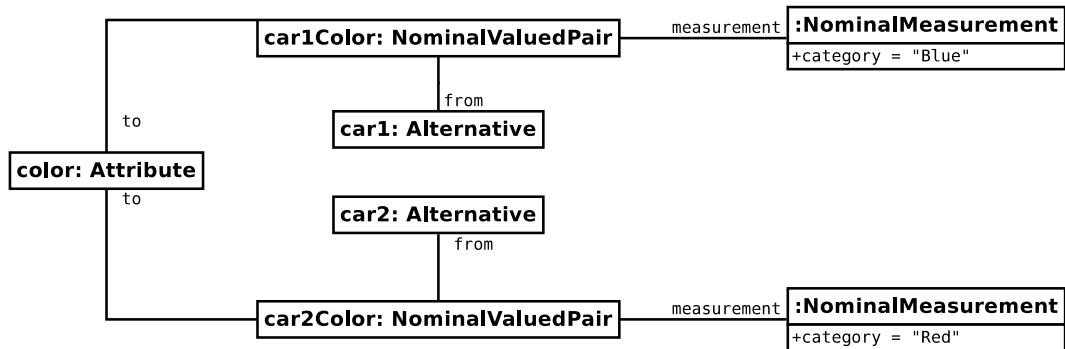


FIGURE 7.8 – Performances des deux voitures sur l'attribut couleur.

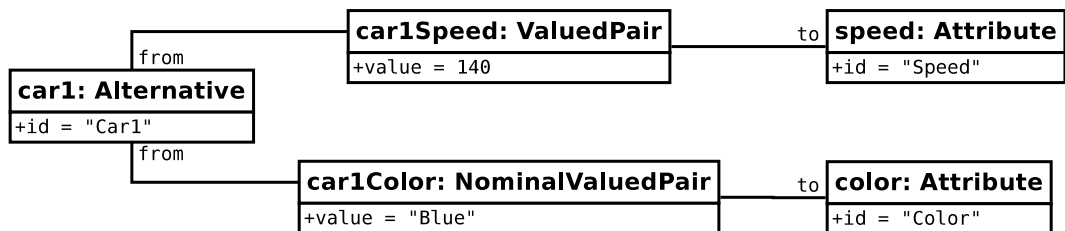


FIGURE 7.9 – Les performances de car1. Les valeurs de car1Speed et car1Color sont en fait des références aux entités correspondantes, mais ont été omises pour plus de clarté.

connexes. La deuxième sous-section montre comment le modèle de données peut être utilisé pour former une méthode en créant un flux connectant des composants individuels.

7.3.1 Encodage des préférences et des performances

Considérons le problème de choix entre deux voitures, car1 et car2. Deux critères interviennent dans cette décision, la vitesse maximale et la couleur. car1 a pour vitesse 140 et est de couleur bleue (« Blue »), car2 a une vitesse maximale de 120 et une couleur rouge (« Red »). La figure 7.8 montre la représentation des couleurs des deux voitures dans le modèle de données, et la figure 7.9 indique avec une notation plus condensée les performances de car1 sur les deux attributs considérés. Le choix entre les deux voitures dépend des préférences du décideur, dont la représentation dépend de la classe de modèles de préférence adoptée. Dans les descriptions suivantes, l'utilisatrice désigne la

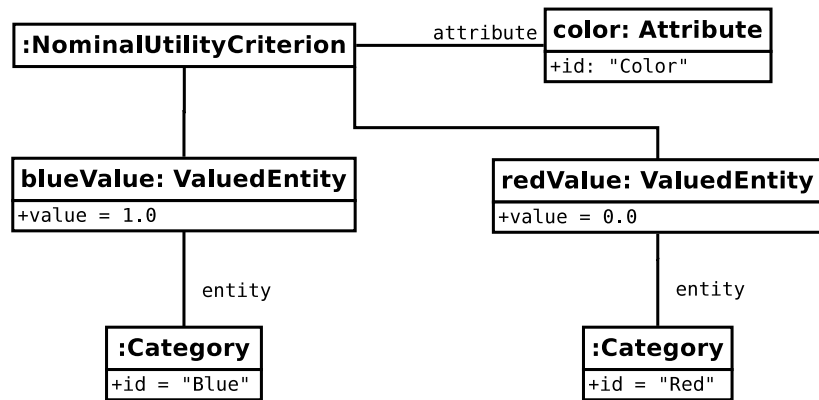


FIGURE 7.10 – Cas 1 : fonctions de valeurs, cas nominal.

personne qui encode les préférences selon le modèle de données.

1. Fonctions de valeurs, cas nominal. L'utilisatrice souhaite encoder une fonction de valeur dans $[0, 1]$ sur l'ensemble des couleurs. Nous sommes dans un cas nominal, car les fonctions attribuent des valeurs à des catégories et non à des valeurs numériques. Il convient simplement dans ce cas d'associer dans le modèle de données les catégories aux valeurs. Supposons que le décideur considère que la couleur bleue mérite une valeur maximale d'utilité (valeur 1) et rouge, minimale (valeur 0). Ceci est représenté par la figure 7.10. L'entité **NominalUtilityCriterion** est connectée à l'attribut nommé Color pour indiquer que ce critère contient des informations de préférence liées à cet attribut. L'entité contient également deux **ValuedEntity**. La première lie la catégorie Blue à la mesure **ExactMeasurement** de valeur 1, la seconde lie la catégorie Red à la valeur 0.
2. Fonctions de valeur, encodage direct. L'utilisatrice ne se soucie pas de la couleur de chaque voiture, mais connaît la valeur, aux yeux du décideur, de la couleur de chaque voiture. Elle souhaite représenter directement la fonction de valeur depuis l'ensemble des alternatives vers l'intervalle $[0, 1]$, sans passer par la couleur de la voiture. Notons que ce cas apparaît naturellement quand un attribut intermédiaire est difficile à trouver (par exemple la beauté d'un paysage ou le confort d'une voiture). La figure 7.11 présente ce cas.
3. Surclassement, avec recodage. Imaginons maintenant que les couleurs des voitures sont connues. Elles sont indiquées dans la figure 7.8. L'utilisatrice souhaite recoder les couleurs (par exemple blue=3, red=10). Pour ce faire, elle souhaite encoder selon le modèle de données la fonction de

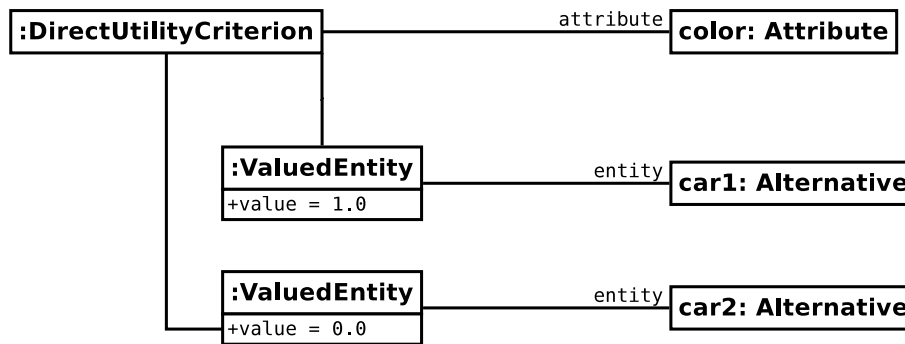


FIGURE 7.11 – Cas 2 : fonctions de valeurs, encodage direct.

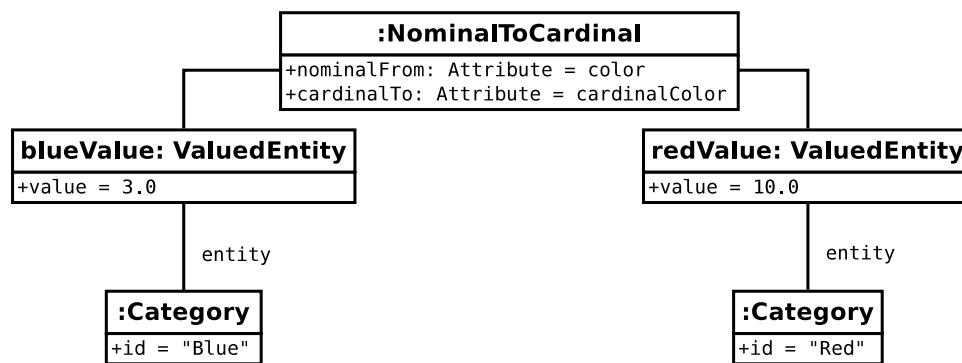


FIGURE 7.12 – Cas 3 : surclassement avec recodage.

recodage, donc l'association entre un ensemble de couleurs possibles et un ensemble de nombres. La figure 7.12 présente l'entité **NominalToCardinal** requise pour le recodage : deux entités valuées sont utilisées. La première indique que la catégorie Blue est associée à la valeur 3 et la seconde associe la catégorie Red à la valeur 10. Observons que cette information, jointe aux performances nominales, définit un nouvel attribut aux performances cardinales. Un composant AMCD pourrait effectuer une telle conversion en prenant en entrée les performances nominales et la fonction de recodage représentée par l'entité **NominalToCardinal**, et produire pour résultat un nouvel attribut (nommé ici `cardinalColor`) associé aux performances numériques correspondantes. Cela lie ce cas au cas suivant.

4. Surclassement, avec recodage et seuils. L'utilisatrice souhaite recoder les couleurs (par exemple `blue=3`, `red=10`), et grâce à ce recodage définir un critère de surclassement associé à des seuils numériques. La figure 7.13 présente ce cas. En effet, un **OutrankingCriterion** doit être lié à un at-

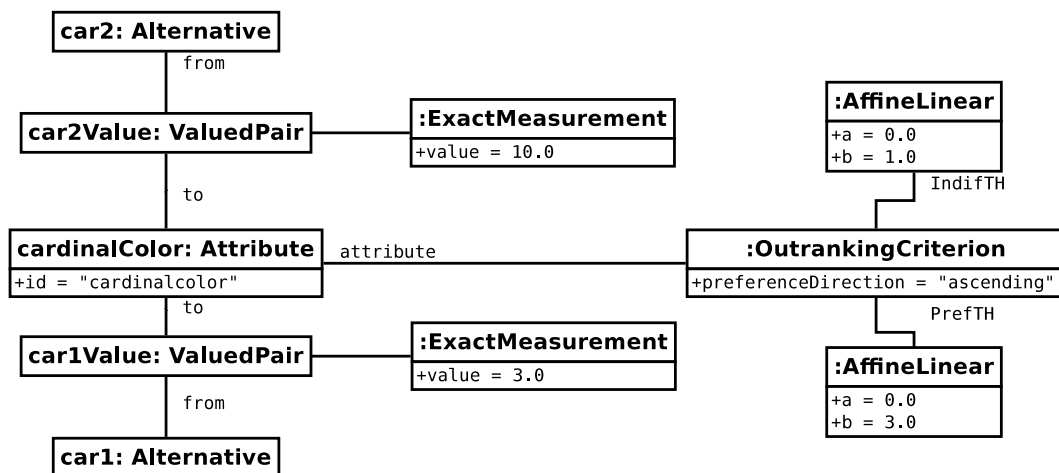


FIGURE 7.13 – Cas 4 : surclassement avec recodage et seuils. Les performances des voitures sur l'attribut couleur ont été recodées en valeurs numériques, associées au nouvel attribut cardinalColor (voir Figures 7.8 et 7.12), ce qui permet de définir les seuils.

tribut associé à des performances numériques, sinon ses valeurs de seuils sont dénuées de sens. C'est pourquoi il faut recoder les performances des voitures préalablement à ces définitions. Le composant défini au cas précédent peut par exemple être utilisé pour ce faire.

5. Fonctions de valeur sur domaine numérique. L'utilisatrice souhaite encoder une fonction de valeur directement sur l'intervalle de valeurs utilisé pour mesurer la vitesse maximale des voitures. La fonction de valeur choisie est linéaire, avec une valeur 0 pour les vitesses inférieures à 110 et 1 pour les vitesses à partir de 160. Ceci est illustré par la figure 7.14.
6. Comparaisons d'alternatives. L'utilisatrice veut encoder des informations de comparaison holistiques de type $\text{car1} \succ \text{car2}$, qui pourront ensuite être utilisées pour inférer un ensemble de modèles de préférence compatibles avec l'information. Seule la paire évaluée binaire doit être encodée, comme le montre la figure 7.15.

Nous omettons la présentation des cas correspondants à l'encodage des informations de préférence sous la forme de matrices d'importances relatives, comme proposé par la méthode AHP : l'encodage est immédiat en utilisant simplement une relation évaluée.

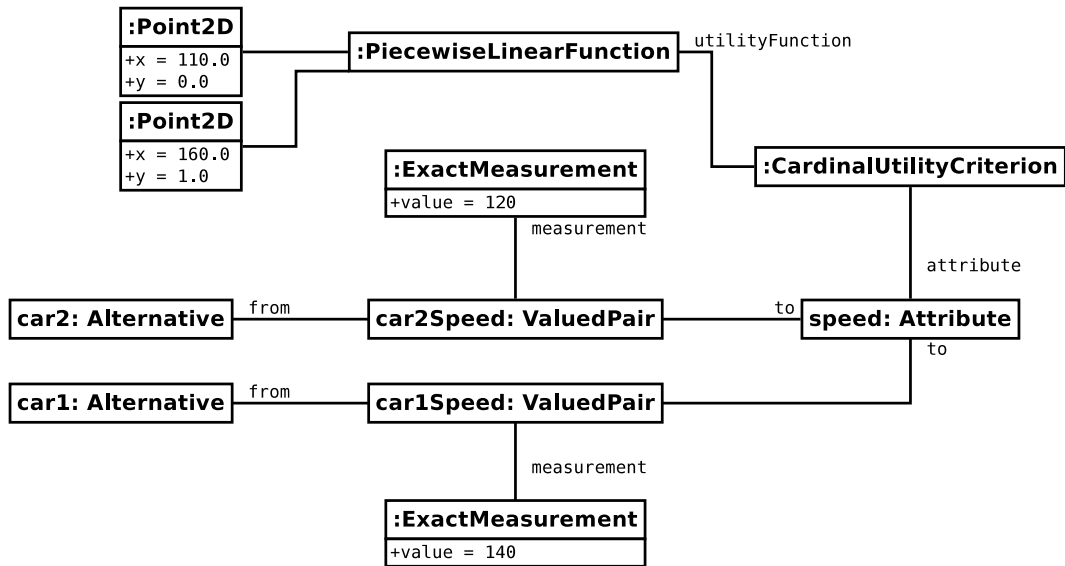


FIGURE 7.14 – Cas 5 : fonctions de valeur sur domaine numérique.



FIGURE 7.15 – Cas 6 : comparaison d'alternatives.

7.3.2 Enchaînement de composants

Un des objectifs du modèle de données est de permettre d'enchaîner facilement des composants dans un flux, ces composants communiquant à travers des interfaces définies de façon stricte. La composition de composants est facilitée s'il n'y a qu'un petit nombre de types différents, et si les types principalement utilisés figurent dans le modèle central. C'est pourquoi nous avons inclus les types représentant des ensembles tels que **AlternativeSet**. Bien qu'il soit facile de définir ces types à partir des autres types grâce aux séquences XML (correspondant au mot clé anglais *sequence*), leur inclusion dans le modèle permet une compatibilité immédiate entre un composant utilisant un de ces types en sortie et un composant utilisant le même type en entrée.

Pour illustrer l'utilisation du modèle en vue de l'enchaînement de composants, considérons le flux de la figure 7.16 visant à construire une relation de surclassement à la façon d'ÉLECTRE. La famille de méthodes ÉLECTRE calcule généralement une relation de concordance, des relations de discordances, et agrège les résultats en une relation de surclassement qui peut ensuite être exploitée. La section 2.2 définit le calcul de concordance, discordance et la relation de surclassement résultante. Dans notre exemple, la relation de concordance est évaluée et la relation de discordance est binaire. Le premier composant de la figure 7.16 calcule la relation de concordance évaluée. Le composant a besoin d'un ensemble d'alternatives, d'un ensemble d'attributs, des performances des alternatives sur les attributs, d'un ensemble de critères (correspondants aux attributs), de poids correspondants aux critères, et de seuils de préférence et d'indifférence pour chaque critère. Dès lors, les critères utilisés en entrée sont des entités **OutrankingCriteria**. La sortie du composant est une relation évaluée, dont les attributs *from* et *to* se rapportent aux alternatives fournies. La relation de concordance évaluée peut alors être envoyée à un composant appelé *cut_relation*, qui prend en plus en entrée un paramètre de seuil de majorité représenté comme un **ExactMeasurement** et transforme la relation évaluée en une relation binaire. Observons que ce composant n'est aucunement spécifique à ÉLECTRE et pourra être utile dans de nombreux contextes.

Le composant calculant la discordance requiert en entrée un ensemble d'alternatives, un ensemble d'attributs, les performances correspondantes, et un ensemble de critères correspondant aux attributs. Chaque critère peut être associé à un seuil de véto. Il est également possible de ne pas associer de valeur de véto à certains critères, auquel cas la condition de véto ne s'applique jamais pour ce critère. Le composant calcule une relation par critère, relations représentées chacune par une entité **ValuedRelation**. Aucun type de notre modèle de données ne permet de représenter cela, il faut donc définir un nouveau type. Il est aisé de définir un type **DiscordanceRelation** contenant une séquence

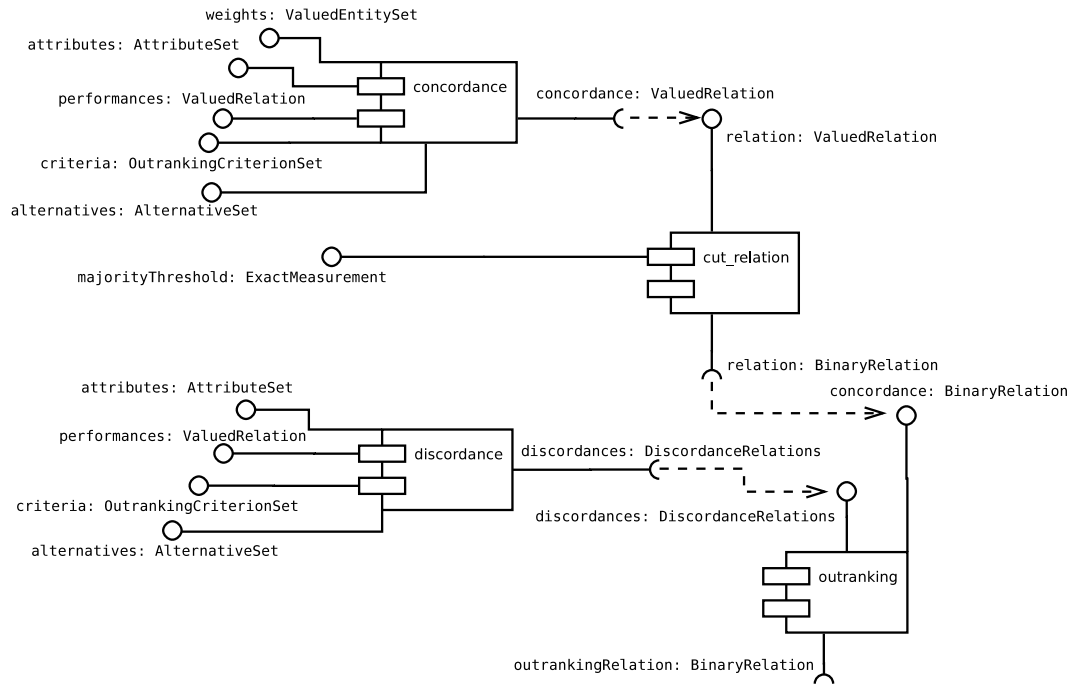


FIGURE 7.16 – Un flux de type ÉLECTRE avec quatre composants : calcul de concordance, coupe d’une relation valuée en une relation nette, calcul de discordance, et calcul de la relation de surclassement.

de paires critère et relation évaluée, car il suffit de réutiliser les types existants et les combiner en une séquence XML. Finalement, la relation de concordance binaire et les relations de discordance évaluées sont envoyées à un composant de surclassement qui les agrège et produit une relation de surclassement sous la forme d'une relation binaire.

7.4 Discussion

Bien que la plupart des méthodes d'AMCD partagent des éléments communs tels que des ensembles d'alternatives, d'attributs, de critères, de nombreuses méthodes manipulent également des concepts propres, et représenter chacun d'entre eux dans un modèle de données standard n'est pas souhaitable. Le modèle que nous proposons ici contient les entités requises pour représenter les données rencontrées couramment en AMCD. Cela permet de contenir la taille du modèle tout en permettant un typage fort. Parce que les entrées et les sorties de différentes méthodes sont différentes, chaque composant devrait définir son propre schéma XML d'entrée et de sortie. Par exemple, les entrées requises pour le composant de concordance d'ÉLECTRE ressemblent aux entrées requises pour le composant de discordance, mais elles ne sont pas identiques. Les schémas XML peuvent inclure des éléments de types définis dans notre modèle, d'autres types standards, ou des types ad hoc. Par exemple, un composant qui produit des PMs devrait réutiliser le type adéquat pour les représenter, proposé par le standard OS (Fourer *et al.*, 2009). Quand un standard adéquat n'existe pas, par exemple parce que le concept à représenter est spécifique à une nouvelle famille de méthodes d'AMCD, une développeuse peut définir son propre type. Un espace de nommage XML propre devrait alors être utilisé pour définir l'identité du nouveau type. Notons que la développeuse peut partager ce nouveau type avec d'autres développeurs de composants travaillant sur des méthodes similaires.

Parce que les compatibilités de type implémentées dans le modèle sont nécessairement imparfaites, il faudra recourir dans certains cas à des convertisseurs de données. Par exemple, une mesure exacte est un cas particulier d'une mesure sous forme d'intervalle, ou d'une fonction affine (dans ce cas, une fonction constante). Mais une entité **ExactValue** n'est pas substituable à une entité **AffineLinear** : pour envoyer une entité du premier type à un composant qui attend une entité du second type, il faudra passer par un convertisseur de données que quelqu'un devra implémenter manuellement. Dans cet exemple, le convertisseur renverra une entité **AffineLinear** représentant la fonction constante égale à la valeur ponctuelle reçue en entrée, donc munie d'un coefficient x égal à zéro. Le logiciel rapidminer, basé sur PMML, se sert d'une

technique similaire en utilisant des convertisseurs pour rendre des types compatibles si nécessaires.

Notons que certains de ces types pourraient être décomposés en entités plus lâchement couplées, par exemple le critère de surclassement pourrait utiliser des seuils codés comme des entités séparées et attachées aux entités critères. Une telle approche permettrait un couplage moins fort et donc des contraintes plus précises sur le typage utilisé par les composants. Cependant, la difficulté d'enchaînement des composants augmente avec le nombre de types à utiliser pour les connecter. C'est pourquoi nous avons décidé d'inclure les seuils avec le critère de surclassement en tant qu'attribut facultatif.

Le modèle de données et les exemples présentés dans ce chapitre sont disponibles en format électronique (voir annexe D, numéro 6).

8 Modélisation et résolution des programmes mathématiques

Ce chapitre, reprenant des informations parues dans un rapport technique du LGI (Cailloux, 2012), fournit des informations techniques détaillées concernant les PMs proposés dans les chapitres précédents. Elles peuvent être utiles pour comprendre l'implémentation qui en est faite dans la bibliothèque J-MCDA (cf. Section 6.2), qui utilise les détails présentés ici. Elles sont également utiles pour tout lecteur qui souhaiterait ré-implémenter ces PMs, car elles donnent des détails importants sur le choix approprié des constantes et sur les stratégies permettant d'éviter les erreurs numériques. De plus, le chapitre présente des optimisations qui permettent de réduire la taille des problèmes à résoudre.

Il nous semble malheureux que les journaux reconnus ne soient en général pas demandeurs de ce niveau de précision (parfois considéré comme abordant des détails techniques inintéressants), ce qui n'incite pas les auteurs à fournir les détails d'implémentation utiles. Il est malheureusement courant de développer les PMs de manière théorique seulement, c'est-à-dire en supposant une précision infinie du solveur, et de laisser au lecteur le soin de chercher comment mettre en œuvre pratiquement le programme résultant. Il nous semble important de fournir une implémentation fonctionnelle des PMs proposés, par exemple sous la forme d'une bibliothèque munie du code source; ainsi que de les exposer suffisamment en détails pour permettre de comprendre l'implémentation et de la reproduire. Dans notre expérience, cette étape pratique peut être aussi périlleuse et couteuse en temps que le développement du PM théorique et devrait donc être valorisée. Cette réflexion devrait être liée à une préoccupation plus générale concernant la reproductibilité des méthodes scientifiques (Ince *et al.*, 2012).

8.1 Erreurs numériques dans les programmes mathématiques

Koch *et al.* (2011) donne une bonne explication des problèmes liés à l'imprécision numérique lors de la résolution de PMs. Le paragraphe suivant est

reproduit de leur section 3.1 (le passage a été traduit par nos soins).

« La plupart des solveurs de PLNEs (...) sont basés sur une arithmétique à virgule flottante et utilisent des tolérances pour vérifier la faisabilité des solutions et décider de leur optimalité. Dans leurs tests de faisabilité, les solveurs utilisent typiquement des tolérances absolues pour les contraintes d'intégralité et des tolérances relatives pour les contraintes linéaires. (...) Les tolérances affectent le temps de résolution et la précision des solutions, généralement de façon opposée (...). Si l'on fixe toutes les variables entières de la solution fournie à la valeur entière la plus proche et que l'on recalcule les valeurs des variables continues en résolvant le PLNE résultant en utilisant des calculs exacts, certaines des solutions obtenues s'avèrent non admissibles considérant des calculs exacts avec tolérance nulle. (...) [Cela signifie] que la solution calculée se trouve hors de la zone admissible décrite par le fichier d'entrée, mais dans la zone admissible étendue créée par la lecture du problème et l'ajout de tolérances. »

Dans ce chapitre, une situation d'erreur numérique désigne une telle situation où le solveur considère certaines valeurs comme satisfaisant certaines contraintes alors que ces valeurs, en calcul exact sans tolérance, ne les satisfèrait pas. Dans les PMs qui nous concernent, de telles erreurs numériques peuvent avoir des conséquences importantes, car ils sont conçus pour satisfaire tous les exemples d'affectation des décideurs. Si aucune mesure particulière n'est prise pour se prémunir de telles erreurs, un solveur pourrait prétendre avoir trouvé une solution admissible mais qui, appliquée au problème original, ne satisfait pas tous les exemples.

L'objet de ce chapitre est de décrire comment modifier certaines contraintes de façon à éviter les situations d'erreurs numériques. Il décrit de plus la façon de choisir les constantes des PMs de façon à s'assurer qu'elles ne réduisent pas artificiellement l'ensemble des solutions admissibles. Enfin, il inclut quelques extensions et optimisations simples permettant de réduire la taille des PMs. Ces optimisations augmentent légèrement la difficulté de compréhension des programmes, c'est pourquoi elles n'ont pas été incluses dans les présentations originales qui se focalisent sur l'exposé de la méthode. Les étiquettes de certaines contraintes affichées dans ce chapitre sont en anglais : elles sont conformes aux noms utilisés dans le code source et affichés dans les fichiers qui peuvent être exportés depuis la bibliothèque J-MCDA (cf. Section 6.2).

8.2 Contexte

Notons que les contraintes sont définies de manière à assurer que le programme est bien défini dans le cas trivial où $n_c = 1$, ce qui peut être utile pour les tests et pour rendre le programme aussi général que possible.

Étant donné un sous-ensemble fini d'alternatives $A \subseteq \mathbb{A}$, les évaluations extrêmes sur X_j sont notées $\underline{g}_j = \min_{a \in A} g_j(a)$ et $\overline{g}_j = \max_{a \in A} g_j(a)$.

Les lecteurs habitués à d'autres présentations d'ÉLECTRE TRI pourraient être surpris par l'absence des profils extrêmes, représentant la limite inférieure de la catégorie la plus mauvaise ou la limite supérieure de la meilleure. Il n'est pas nécessaire de définir ces valeurs dans notre définition de la méthode, mais elles sont parfois utilisées dans des définitions différentes, telle que la présentation originale de la méthode. Les définitions sont cependant équivalentes. Il suffit de définir le pire et le meilleur profil avec les évaluations les pires et les meilleures possibles, $g_j(l^-(C_1)) = \underline{g}_j$, $g_j(l^+(C_{n_c})) = \overline{g}_j$, pour obtenir une représentation du modèle adéquate pour la définition originale.

8.2.1 Constantes utilisées dans les programmes

Les constantes utilisées dans les programmes expliqués ci-dessous sont définies comme suit.

- \underline{g}_j et \overline{g}_j , la pire et la meilleure performance sur le critère j . Définir \underline{g}_j inférieure à la pire performance sur le critère j est également acceptable, il faut seulement que cette notation représente une borne inférieure sur les performances pouvant être atteintes par une alternative quelconque sur ce critère. La remarque s'applique de manière similaire à \overline{g}_j .
- $\forall j \in J : \delta_j$, une valeur inférieure à la différence minimale entre deux performances différentes sur j ($\forall j \in J : \min_j = \min_{a, a' \in A} |g_j(a) - g_j(a')|$) divisée par n_c . Cela laisse assez d'espace pour insérer $n_c - 1$ profils dans l'intervalle $[g_j(a), g_j(a')]$ correspondant à n'importe quelle paire de performances distinctes sur j atteintes par des alternatives, tout en séparant chaque valeur de profil par δ_j . S'il n'y a pas deux performances différentes pour un critère donné, δ_j est défini arbitrairement à 1. Cette valeur est utilisée pour transformer des contraintes d'inégalité stricte en contraintes larges dans les contraintes $(b, \text{increase})$, (C, floor) , $(C, \text{ceiling})$, (V, floor) , $(V, \text{ceiling})$, (V, used) . L'implémentation utilise $\delta_j = \min_j / (k + 1)$.
- $\forall j \in J : \delta_j^b$, la marge associée aux profils, séparation maximale entre les profils extrêmes (resp. meilleur et pire) et les performances extrêmes (resp. \overline{g}_j et \underline{g}_j). Pour permettre aux profils successifs un écart d'au moins δ_j entre eux et avec les performances extrêmes, il faut définir δ_j^b de manière telle que $\delta_j^b \geq (k - 1)\delta_j$. Ceci est nécessaire pour laisser la possibilité aux profils d'être tous supérieurs à la meilleure performance, ou inférieurs à la pire performance. L'implémentation utilise $\delta_j^b = k\delta_j$. Cette constante est utilisée pour définir les bornes des profils extrêmes b_1 et b_{k-1} , et indirectement dans les contraintes impliquant M_j .

- $\forall j \in J : M_j = \overline{g_j} - \underline{g_j} + \delta_j^b$, un facteur de mise à l'échelle représentant la différence maximale de performance possible sur le critère j entre un profil et une alternative. Les contraintes (C, floor) , $(C, \text{ceiling})$, (V, floor) , $(V, \text{ceiling})$, (V, used) utilisent cette valeur.
- δ_λ , une valeur positive arbitrairement petite, utilisée dans les contraintes $(\text{support}, \text{ceiling})$ et $(\text{support}, \text{ceiling-v})$. L'implémentation utilise $\delta_\lambda = 0,001$.
- δ_j^v , une valeur utilisée dans la définition des variables de véto et dans les contraintes $(V, \text{ceiling})$, qui doit être supérieure à δ_j . Dans l'implémentation, $\delta_j^v = \delta_j * 2$.

8.3 ICL

Rappelons l'objectif du programme ICL. Soient un ensemble d'alternatives $X \subseteq \mathbb{A}$ utilisées comme exemples d'affectation, un ensemble d'indices de critères \mathcal{J} , les évaluations des alternatives exemples $a_j, \forall a \in X, j \in \mathcal{J}$, des catégories $\mathcal{C} = \{C_1, \dots, C_k\}$, un ensemble de profils associés L , un ensemble de décideurs \mathcal{T} , des exemples d'affectation $E^t, \forall t \in \mathcal{T}$. Le programme ICL détermine les performances des profils $l \in L$, partagés entre les décideurs, et des valeurs individuelles de poids $w_j^t, \forall j \in \mathcal{J}$ et de seuils de majorité λ^t , pour chaque décideur $t \in \mathcal{T}$, satisfaisant tous les exemples d'affectation. Rappelons qu'un profil est noté simplement l lorsque les catégories que le profil sépare ne sont pas importantes, ou $l^-(C)$ ou $l^+(C)$ pour représenter la limite inférieure ou supérieure de la catégorie C .

8.3.1 Contraintes

Cette section ne consiste pas en une nouvelle présentation complète du programme ICL : les justifications et explications des contraintes pour lesquelles aucune information technique n'est à préciser ne sont pas reprises. Nous ne considérons que les parties concernant les contraintes pour lesquelles des précisions sont apportées. Le programme est toutefois rappelé dans son entièreté en fin de section.

Séparation des profils

Les contraintes (profils croissants) assurent que les profils sont ordonnés correctement. Les contraintes suivantes leur correspondent mais assurent que les profils sont séparés au moins par une valeur δ_j . $\forall C \in \mathcal{C} \setminus \{C_1, C_n\}, j \in \mathcal{J}$:

$$l_j^-(C) \leq l_j^+(C) - \delta_j. \quad (b, \text{increase})$$

S'assurer que les profils sont séparés d'une valeur supérieure à la tolérance du solveur permet d'éviter les erreurs numériques qui surviendraient si un profil était supérieur à un autre d'une marge très faible. Les séparer de δ_j n'empêche aucunement qu'ils soient sémantiquement égaux, c'est-à-dire que la valeur de plusieurs profils s'intercale au même endroit dans l'ordre des performances sur ce critère : la définition de δ_j assure que la valeur est suffisamment petite. Observons qu'il est alors possible que deux profils aient des valeurs sémantiquement égales sur tous les critères. Cela se produit ssi une catégorie au moins ne contient aucune alternative de l'ensemble d'exemples, donc une condition suffisante pour que cela ne se produise pas est que $\forall C \in \mathcal{C} : \exists t \in \mathcal{T}, a \in X \mid (a \xrightarrow{t} [\underline{C}, \overline{C}]) \in E^t$.

Bornes des profils

Il est nécessaire pour que les constantes soient bien définies de délimiter les valeurs maximales des profils (si $n_C \geq 2$). $\forall j \in \mathcal{J}$:

$$\begin{aligned} \underline{g}_j - \delta_j^b &\leq l^+(C_1), \\ l^-(C_{n_C}) &\leq \overline{g}_j + \delta_j^b. \end{aligned}$$

Concordance binaire

Considérons la définition de $K_j(a, l)$ (reprise de (concordance binaire) page 75), qui indique si $a_j \geq l_j$. $\forall j \in \mathcal{J}, a \in X, l \in L$:

$$\frac{1}{M_j}(a_j - l_j) < K_j(a, l), \quad (8.1)$$

$$K_j(a, l) \leq \frac{1}{M_j}(a_j - l_j) + 1. \quad (8.2)$$

De façon à transformer (8.1) en une contrainte large, la valeur δ_j peut être utilisée. La façon dont elle est définie assure qu'ajouter cette valeur à la contrainte ne réduit pas l'ensemble de valeurs acceptables pour l_j . Ainsi, $\forall j \in \mathcal{J}, a \in X, l \in L$:

$$\frac{1}{M_j + \delta_j}(a_j - l_j + \delta_j) \leq K_j(a, l). \quad (C, \text{floor})$$

Les contraintes (8.2) sont sujettes à des erreurs numériques. Le solveur pourrait trouver une solution telle que $l_j = a_j + \varepsilon$, pour une alternative a , un critère j , un profil l , avec ε une petite valeur positive, et considérer $K_j(a, l) = 1$ bien que $K_j(a, l)$ devrait valoir zéro car la valeur du profil est supérieure à la performance de l'alternative. Ceci pourrait arriver car il est possible que $K_j(a, l) = 1$

satisfasse la contrainte (8.2) exigeant $K_j(a, l) \leq \frac{-\varepsilon}{M_j} + 1$ à cause de la tolérance du solveur. Pour éviter cette situation, les contraintes (8.2) doivent être transformées :

$$K_j(a, l) \leq \frac{1}{M_j + x_j}(a_j - l_j - x_j) + 1. \quad (C, \text{ ceiling})$$

En notant t la tolérance utilisée par le solveur pour les valeurs entières, $\frac{x_j}{M_j + x_j}$ devrait valoir au moins t , donc $x_j \geq (M_j + x_j)t$. Ceci rend la contrainte plus forte : il est maintenant assuré que $a_j < l + x_j \Rightarrow K_j(a, l) = 0$. La modification empêche des situations où $l_j < a_j < l_j + x_j$, donc, empêche de choisir la performance du profil de telle sorte que $\exists a \in X \mid a_j - x_j < l_j < a_j$. Il faut fixer x_j suffisamment petit pour éviter de réduire l'espace des solutions admissibles. Une solution pratique est de définir $x_j = \delta_j$. Pour que cette solution soit valide, il faut $\frac{\delta_j}{M_j + \delta_j} \geq t, \forall j \in J$. Par exemple, le solveur CPLEX solver version 12.3 utilise par défaut $t = 10^{-5}$. Ceci implique que la stratégie proposée ne s'applique pas si le ratio entre la plus petite différence et la plus grande différence est trop faible. Une solution à ce problème est de ré-échelonner les données. Une stratégie alternative est d'éviter d'utiliser les valeurs données par le solveur pour l_j , plutôt que de changer la contrainte (8.2). Les valeurs des profils peuvent en effet être déduites des valeurs de $K_j(a, l)$. Ceci pourrait constituer une meilleure stratégie étant donné qu'elle n'impose pas de restriction aux données d'entrée, mais elle n'est pas implémentée pour le moment.

Traitement adéquat des coalitions

Les contraintes de définition du support s'expriment comme suit (cf. (support plancher) et (support plafond)), $\forall t \in \mathcal{T}, \forall (a \xrightarrow{t} [\underline{C}, \overline{C}]) \in E^t$:

$$\sum_{j \in \mathcal{J}} \Psi_j^t(a, l^-(\underline{C})) \geq \lambda^t \text{ si } \underline{C} \neq C_1, \quad (\text{support, floor})$$

et

$$\sum_{j \in \mathcal{J}} \Psi_j^t(a, l^+(\overline{C})) \leq \lambda^t - \delta_\lambda \text{ si } \overline{C} \neq C_n. \quad (\text{support, ceiling})$$

La deuxième a été transformée en inégalité large grâce à la constante δ_λ .

Les contraintes (support, floor) sont sujettes aux erreurs numériques. Il est possible (et cela a été observé avec CPLEX) que $\sum_{j \in \mathcal{J}} \Psi_j^t(a, l^-(\underline{C})) = \lambda^t - \varepsilon$, pour une alternative a , et profil $l^-(\underline{C})$ donnés, avec une petite valeur positive ε , de telle sorte que le solveur considère la contrainte satisfaite alors qu'elle ne l'est strictement parlant pas. Modifier la contrainte en lui additionnant une petite valeur positive δ'_λ n'est pas une solution adéquate parce que cela restreindrait l'espace des solutions admissibles. En particulier, cela exclurait les

solutions où $\lambda = 0,5$ impliquant deux coalitions suffisantes disjointes. De plus, le problème appelle une solution plus générale. Même hors de toute considération liée à des PMs, de l'imprécision peut être introduite lors de la lecture des poids depuis un fichier, ou lors de l'obtention des poids depuis toute autre source de précision non infinie. L'imprécision peut aussi provenir du calcul de la somme des poids lui-même : la somme calculée pourrait être très faiblement inférieure à la valeur λ alors que la somme réelle serait supérieure ou égale à λ . La solution que nous avons adoptée pour résoudre généralement cette classe de problèmes est la suivante. Lorsque la somme des poids est comparée au seuil de majorité λ , nous utilisons une tolérance ε qui peut être configurée dans notre bibliothèque. Si $\sum_{j \in J | a_j \geq l_j} w_j \geq \lambda^t - \varepsilon$, nous considérons que l'alternative surclasse le profil, comme si la situation était telle que $\sum_{j \in J | a_j \geq l_j} w_j \geq \lambda^t$. Ceci requiert de s'assurer que les vraies valeurs des poids (en considérant une précision infinie) sont telles qu'aucune coalition n'a de valeurs dans $[\lambda - \varepsilon, \lambda[$, sinon l'introduction de la tolérance modifie le résultat.

Revenons au PM. Pour éviter les erreurs numériques, nous modifions l'interprétation des résultats plutôt que les contraintes incriminées. Choisissons une valeur $\varepsilon < \delta_\lambda$ (nous suggérons $\varepsilon = \delta_\lambda/2$), et interprétons les coalitions dans ÉLECTRE TRI comme indiqué ci-dessus. L'introduction de cette tolérance permet d'interpréter le résultat correctement car les contraintes (support, ceiling) assurent que soit la coalition est inférieure ou égale à $\lambda^t - \delta_\lambda$, soit elle devrait être considérée comme une coalition gagnante. Ceci résout le problème pour autant que $\varepsilon > t$, en notant t la tolérance utilisée par le solveur pour les contraintes linéaires (CPLEX version 12.3 utilise $t = \frac{1}{10^6}$ par défaut).

Reste un problème lié à l'utilisation de cette stratégie. Les valeurs des poids pris de la solution du solveur doivent être interprétés à l'aide d'une valeur ε appropriée, ce que les utilisateurs pourraient ne pas savoir. Une solution à ce problème serait de recalculer des poids et un seuil de majorité équivalents (ayant les mêmes coalitions suffisantes) et ne posant pas de problème d'imprécision. La bibliothèque montrerait alors à l'utilisateur les valeurs transformées plutôt que les valeurs directement issues du solveur. Ceci n'est pas implémenté actuellement.

8.3.2 Réduction de la taille du programme mathématique

Les variables $K_j(a, l)$ et $\Psi_j^t(a, l)$ sont, dans ce qui précède, définies pour toute alternative $a \in X$ et profil $l \in L$. Or, elles ne sont pas toutes requises : ce sont des variables techniques, utilisées pour s'assurer de la satisfaction des exemples d'affectation des décideurs. Il n'est nécessaire de les définir que dans la mesure où elles sont utilisées dans d'autres contraintes, donc dans la mesure où elles

interviennent dans au moins un exemple d'affectation. Il est ainsi possible de réduire de manière importante le nombre de variables et de contraintes du PM.

Notons

$$P^t = \left\{ (a, l) \in X \times L \mid \exists \underline{C}, \overline{C} \in \mathcal{C}, (a \xrightarrow{t} [\underline{C}, \overline{C}]) \in E^t \mid l = l^-(\underline{C}) \vee l = l^+(\overline{C}) \right\}$$

et

$$P = \bigcup_{t \in \mathcal{T}} P^t.$$

Les variables $K_j(a, l)$ doivent être définies $\forall (a, l) \in P, j \in \mathcal{J}$. Les variables $\Psi_j^t(a, l)$ doivent être définies $\forall t \in \mathcal{T}, (a, l) \in P^t, j \in \mathcal{J}$. Le nombre de contraintes (C, floor) , $(C, \text{ceiling})$ servant à définir les variables $K_j(a, l)$ et (σ, floor) , $(\sigma, \text{ceiling } C)$, $(\sigma, \text{ceiling } w)$ (correspondant aux contraintes (concordance pondérée 1) et (concordance pondérée 2)) servant à définir les variables $\Psi_j^t(a, l)$ peut dès lors également être réduit.

Les contraintes (somme poids), correspondant à (weights, sum), ne sont pas modifiées. Voici toutes les contraintes composant le programme modifié.

$\forall t \in \mathcal{T}$:

$$\sum_{j \in \mathcal{J}} w_j^t = 1. \quad (\text{weights, sum})$$

$\forall C \in \mathcal{C} \setminus \{C_1, C_n\}, j \in \mathcal{J}$:

$$l_j^-(C) \leq l_j^+(C) - \delta_j. \quad (b, \text{increase})$$

$\forall (a, l) \in P, j \in \mathcal{J}$:

$$\frac{1}{M_j + \delta_j} (a_j - l_j + \delta_j) \leq K_j(a, l), \quad (C, \text{floor})$$

$$K_j(a, l) \leq \frac{1}{M_j + \delta_j} (a_j - l_j - \delta_j) + 1. \quad (C, \text{ceiling})$$

$\forall t \in \mathcal{T}, (a, l) \in P^t, j \in \mathcal{J}$:

$$w_j^t + K_j(a, l) - 1 \leq \Psi_j^t(a, l) \quad (\sigma, \text{floor})$$

$$\Psi_j^t(a, l) \leq K_j(a, l) \quad (\sigma, \text{ceiling } C)$$

$$\Psi_j^t(a, l) \leq w_j^t. \quad (\sigma, \text{ceiling } w)$$

$\forall t \in \mathcal{T}, \forall (a \xrightarrow{t} [\underline{C}, \overline{C}]) \in E^t$:

$$\sum_{j \in \mathcal{J}} \Psi_j^t(a, l^-(\underline{C})) \geq \lambda^t \text{ si } \underline{C} \neq C_1, \quad (\text{support, floor})$$

$$\sum_{j \in \mathcal{J}} \Psi_j^t(a, l^+(\overline{C})) \leq \lambda^t - \delta_\lambda \text{ si } \overline{C} \neq C_n. \quad (\text{support, ceiling})$$

8.4 ICLV

Pour rappel, le programme ICLV étend le programme d'inférence ICL en cherchant un modèle de préférence incluant éventuellement des vétos. Les valeurs des seuils de vétos sont représentées par des variables continues $\nu_j^{\text{BM}}(C)$, associées à chaque critère $j \in \mathcal{J}$ et catégorie $C \in \mathcal{C} \setminus \{C_1\}$. Ces variables représentent les seuils de veto au sens indiqué à la section 2.2.3, soit $\nu_j^{\text{BM}}(C) = l_j^-(C) - v_j$, donc une valeur en-dessous de laquelle l'alternative est interdite d'accès en catégorie C ou mieux.

Ne pas utiliser de seuil de veto est équivalent à fixer $\nu_j^{\text{BM}}(C) < \underline{g}_j$. Les variables $\nu_j^{\text{BM}}(C)$ doivent donc être comprises dans $[\underline{g}_j - \delta_j^v, l_j^-(C)]$.

L'implémentation utilise des variables $v_j = l_j^-(C) - \nu_j^{\text{BM}}(C)$, car cette définition est la définition générale du veto s'appliquant également en dehors du contexte du tri et que cette définition rend l'implémentation (en cas de seuils de veto constants) nettement plus économique. Notons que le cas des seuils de veto non constants n'a pas encore été implémentée. De façon à clarifier la relation entre l'implémentation et l'exposé de la méthode, cette section donne les deux formes des contraintes, celles avec la définition du veto utilisant $\nu_j^{\text{BM}}(C)$ et celles de l'implémentation utilisant v_j .

Dans la définition utilisée dans l'implémentation, une alternative a ne peut atteindre une catégorie C si $l_j^-(C) - a_j > v_j$, ce qui est équivalent à $a_j < \nu_j^{\text{BM}}(C)$. Dès lors, ne pas utiliser un seuil de veto revient à fixer $v_j \geq l_j^-(C) - \underline{g}_j$, et les variables v_j doivent être définies dans $[0, l_j^-(C) - \underline{g}_j + \delta_j^v]$, pour une variable correspondant à la catégorie C . Comme nous utilisons des seuils de veto constants (indépendants de la catégorie), nous pouvons fixer simplement les bornes supérieures à $v_j \leq M_j + \delta_j^v$.

8.4.1 Contraintes

Notons que les contraintes (veto, plafond) correspondent simplement à la borne inférieure à zéro des variables v_j .

L'ordre sur les seuils de veto est respecté grâce aux contraintes suivantes (cf. (veto, ordre)), $\forall C_h \in \mathcal{C} \setminus \{C_1, C_n\}, j \in \mathcal{J}$:

$$\nu_j^{\text{BM}}(C_h) \leq \nu_j^{\text{BM}}(C_{h+1}). \quad (\text{veto, ordre})$$

Ces contraintes ne doivent pas être définies pour des seuils de veto constants. Les contraintes équivalentes pour des seuils de veto $v_j(C_h)$ qui dépendraient de la catégorie seraient les suivantes :

$$v_j(C_h) - v_j(C_{h-1}) \leq l_j^-(C_h) - l_j^-(C_{h-1}). \quad (v, \text{order})$$

8 Modélisation et résolution des programmes mathématiques

Les variables binaires $V_j(a, C)$ étaient définies comme suit (cf. (véto binaire)), de manière à garantir que $V_j(a, C) = 1 \Leftrightarrow a_j < \nu_j^{\text{BM}}(C) : \forall a \in X, C \in \mathcal{C} \setminus \{C_1\}, j \in \mathcal{J}$,

$$\frac{\nu_j^{\text{BM}}(C) - a_j}{M} \leq V_j(a, C) \leq \frac{\nu_j^{\text{BM}}(C) - a_j - \varepsilon}{M} + 1. \quad (\text{véto binaire})$$

(Une erreur s'est glissée à ce stade de l'explication dans le rapport technique, qui implémente plutôt $V_j(a, C) = 1 \Leftrightarrow a_j \leq \nu_j^{\text{BM}}(C)$, erreur qui ne prête pas à conséquence étant donné que ce qui suit garantit $a_j \neq \nu_j^{\text{BM}}(C)$.)

Elles sont implémentées comme suit :

$$\frac{l_j^-(C) - a_j - v_j + \delta_j}{M_j + \delta_j} \leq V_j(a, C), \quad (V, \text{ floor})$$

$$V_j(a, C) \leq \frac{l_j^-(C) - a_j - v_j - \delta_j}{2M_j + \delta_j^v + \delta_j} + 1. \quad (V, \text{ ceiling})$$

Les dénominateurs sont choisis de façon à borner les fractions, l'une supérieurement par 1, l'autre inférieurement par -1, afin de permettre le raisonnement proposé pour les contraintes (concordance binaire). Les bornes sont garanties car $l_j^-(C) - a_j - v_j \in [-2M_j - \delta_j^v, M_j]$. Concernant le numérateur, l'ajout de δ_j dans (V, floor) permet d'éviter les erreurs numériques : (V, floor) et (V, ceiling) assurent qu'il n'existe pas $l_j^-(C) - a_j = v_j$ (à condition que $\frac{\delta_j}{2M_j + \delta_j^v + \delta_j} \geq t$, avec t la tolérance du solveur), ce qui garantit une marge de sécurité contre les imprécisions.

Les contraintes (support plancher avec véto) et (support plafond avec véto) modifiant les contraintes (support plancher) et (support plafond) ne sont pas modifiées.

La variable binaire V_j doit valoir 1 ssi un véto est utilisé pour ce critère. Grâce à la fonction objectif, il suffit de contraindre V_j à un si nécessaire, ce qui peut se faire à l'aide de ces contraintes, $\forall j \in J$:

$$V_j \geq \frac{\nu_j^{\text{BM}}(C_{nc}) - \underline{g}_j + \delta_j}{M_j + \delta_j},$$

ou de manière équivalente :

$$V_j \geq \frac{l_j^-(C_{nc}) - \underline{g}_j + \delta_j - v_j(C_{nc})}{M_j + \delta_j}. \quad (V, \text{ used})$$

Cette formulation est plus économique que la contrainte originale (véto utilisé) ($V_j \geq V_j(a, C), \forall a, C$). Elle garantit : $[\exists C \in \mathcal{C} \mid \nu_j^{\text{BM}}(C) \geq \underline{g}_j] \Rightarrow [\nu_j^{\text{BM}}(C_{nc}) \geq \underline{g}_j] \Rightarrow [\nu_j^{\text{BM}}(C_{nc}) - \underline{g}_j + \delta_j > 0] \Rightarrow V_j = 1$.

8.4.2 Vétos et critères de poids nul

Une question se pose concernant le choix d'autoriser ou d'interdire les vétos associés aux critères de poids nul. Nous avons choisi d'autoriser de telles solutions, comme expliqué à l'annexe C. Dans le cas contraire, il faudrait ajouter des contraintes appropriées pour interdire cette situation.

8.4.3 Programme mathématique réduit

Le programme peut être réduit de la même façon que le précédent, en évitant l'usage des variables $V_j(a, C)$ n'intervenant dans aucun exemple.

Voici toutes les contraintes composant le programme modifié. L'objectif est

$$\min \sum_{j \in \mathcal{J}} V_j.$$

$\forall t \in \mathcal{T} :$

$$\sum_{j \in \mathcal{J}} w_j^t = 1. \quad (\text{weights, sum})$$

$\forall C \in \mathcal{C} \setminus \{C_1, C_n\}, j \in \mathcal{J} :$

$$l_j^-(C) \leq l_j^+(C) - \delta_j. \quad (b, \text{increase})$$

$\forall (a, l) \in P, j \in \mathcal{J} :$

$$\frac{1}{M_j + \delta_j} (a_j - l_j + \delta_j) \leq K_j(a, l), \quad (C, \text{floor})$$

$$K_j(a, l) \leq \frac{1}{M_j + \delta_j} (a_j - l_j - \delta_j) + 1. \quad (C, \text{ceiling})$$

$\forall t \in \mathcal{T}, (a, l) \in P^t, j \in \mathcal{J} :$

$$w_j^t + K_j(a, l) - 1 \leq \Psi_j^t(a, l) \quad (\sigma, \text{floor})$$

$$\Psi_j^t(a, l) \leq K_j(a, l) \quad (\sigma, \text{ceiling } C)$$

$$\Psi_j^t(a, l) \leq w_j^t. \quad (\sigma, \text{ceiling } w)$$

$\forall C_h \in \mathcal{C} \setminus \{C_1, C_n\}, j \in \mathcal{J} :$

$$v_j(C_h) - v_j(C_{h-1}) \leq l_j^-(C_h) - l_j^-(C_{h-1}). \quad (v, \text{order})$$

$\forall t \in \mathcal{T}, (a \xrightarrow{t} [\underline{C}, \overline{C}]) \in E^t, \underline{C} \neq C_1, j \in \mathcal{J} :$

$$\frac{l_j^-(\underline{C}) - a_j - v_j + \delta_j}{M_j + \delta_j} \leq V_j(a, \underline{C}), \quad (V, \text{floor})$$

$$V_j(a, \underline{C}) \leq \frac{l_j^-(\underline{C}) - a_j - v_j - \delta_j}{2M_j + \delta_j^v + \delta_j} + 1. \quad (V, \text{ceiling})$$

$\forall t \in \mathcal{T}, \forall (a \xrightarrow{t} [\underline{C}, \overline{C}]) \in E^t :$

$$\sum_{j \in \mathcal{J}} \Psi_j^t(a, l^-(\underline{C})) \geq \lambda^t + \sum_{j \in \mathcal{J}} V_j(a, l^-(\underline{C})) \text{ si } \underline{C} \neq C_1, \quad (\text{support, floor-v})$$

$$\sum_{j \in \mathcal{J}} \Psi_j^t(a, l^+(\overline{C})) \leq \lambda^t - \delta_\lambda + \sum_{j \in \mathcal{J}} V_j(a, l^+(\underline{C})) \text{ si } \overline{C} \neq C_n. \quad (\text{support, ceiling-v})$$

$\forall j \in J :$

$$V_j \geq \frac{l_j^-(C_{nc}) - \underline{g}_j + \delta_j - v_j(C_{nc})}{M_j + \delta_j}. \quad (V, \text{ used})$$

Les constantes $\underline{g}_j, \overline{g}_j, M_j, \delta_j, \delta_j^b, \delta_j^v$ et δ_λ sont définies à la section 8.2.1. Les variables $K_j(a, l), V_j(a, C)$ et V_j sont binaires, l_j réelles, avec $\underline{g}_j - \delta_j^b \leq l^+(C_1)$ et $l_j^-(C_{nc}) \leq \overline{g}_j + \delta_j^b, w_j^t, \Psi_j^t(a, l)$ sont réelles et non négatives, $\lambda^t \in [1/2, 1], v_j \in [0, M_j + \delta_j^v]$.

8.5 Contraintes de portefeuilles

Le PM peut également inclure des contraintes sur la taille des catégories, définies par (n implique affectation), (somme n), (taille catégorie) page 133. Ces contraintes peuvent être remplacées par les suivantes, plus économes.

Définissons les variables binaires $n^{\geq}(a, C), \forall a \in A, C \in \mathcal{C} \setminus C_1$. La variable $n^{\geq}(a, C)$ vaut 1 ssi $a \succeq C$, donc ssi $\sum_{j \in \mathcal{J}} \Psi_j(a, l^-(C)) \geq \lambda$. $\forall a \in A, C \in \mathcal{C} \setminus C_1 :$

$$\begin{aligned} \sum_{j \in \mathcal{J}} \Psi_j(a, l^-(C)) &\geq \lambda + (n^{\geq}(a, C) - 1), \\ \sum_{j \in \mathcal{J}} \Psi_j(a, l^-(C)) &\leq \lambda + n^{\geq}(a, C) - \delta_\lambda. \end{aligned} \quad (8.3)$$

Ces contraintes garantissent que $a \succeq C \Leftrightarrow n^{\geq}(a, C) = 1$. La remarque de la section 8.3.1 concernant le traitement des coalitions s'applique ici et permet de contourner les risques d'imprécision numérique liés à la contrainte (8.3).

Ces variables sont utilisées pour assurer la satisfaction des contraintes de portefeuille. $\forall \langle C_h, \underline{\eta}, \overline{\eta}, P \rangle \in N :$

$$\underline{\eta} \leq \sum_{a \in A} n^{\geq}(a, C_h) P(a) - n^{\geq}(a, C_{h+1}) P(a) \leq \overline{\eta},$$

avec $\forall a \in A : n^{\geq}(a, C_1) = 1$ et $n^{\geq}(a, C_{nc+1}) = 0$. Les variables concernant les catégories extrêmes C_1 et C_{nc+1} ne doivent pas être définies, leurs valeurs étant constantes. Il s'agit simplement ici d'une notation particulière.

Cette formulation offre un avantage important lorsque les contraintes de portefeuilles ne s'expriment que sur les catégories les meilleures. Dans ce cas, il n'est pas nécessaire de définir les variables $n(a, C)$ concernant les catégories non concernées par les contraintes. Cela implique également qu'il devient inutile de définir les variables correspondantes $\Psi_j(a, l^-(C))$ et K_j nécessaires à la définition des $n(a, C)$. Par exemple, l'illustration donnée à la section 5.4 n'utilise que des contraintes de portefeuille concernant C_3 , ce qui conduirait à définir, avec la formulation proposée initialement, beaucoup de variables et de contraintes superflues. Dans les applications pratiques, il est donc probable que la formulation proposée ici soit nettement plus économe que la formulation initiale.

Conclusions et perspectives

To a scientist, a thought experiment is an argument that you can run through in your head, after which you understand what's going on so well that there's no need to do a real experiment, which is of course a great saving in time and money and prevents you from getting embarrassingly inconvenient results.

(Terry Pratchett, The science of Discworld)

9 Conclusions et perspectives

9.1 Conclusions

Dans ce travail, nous avons proposé plusieurs méthodes d'élicitation visant à construire des modèles de tri, et plus précisément des modèles de type MR Sort. Ces méthodes s'appuient sur le principe de la désagrégation.

Nous avons décrit au chapitre 2 différentes classes de modèles de tri, avec pour certaines leur axiomatique, ce qui permet de les comparer et de situer la classe de modèle MR Sort par rapport aux autres classes de modèles de tri. Ces résultats axiomatiques indiquant les relations d'inclusion entre classes de modèles ouvrent une perspective : dans une approche d'élicitation utilisant la désagrégation, on pourrait chercher à éliciter des modèles simples, et passer aux modèles plus complexes seulement si nécessaire, selon le principe du rasoir d'Ockham. Un modèle peut être dit plus simple s'il appartient à une classe de modèles moins générale, dans le sens où il faut moins de paramètres pour le définir.

Par exemple, dans le contexte du tri, on voit que la classe NCSM sans véto est incluse dans la classe des modèles de tri à fonction croissante et à seuils. Dès lors, une approche de désagrégation pourrait commencer par chercher un modèle de type NCSM sans véto. Si les exemples ne peuvent être représentés dans ce modèle, même après utilisation d'outils de résolution des inconsistances, on peut chercher à représenter les exemples dans le modèle plus général. Cette démarche serait équivalente à celle consistant à tenter de modéliser un phénomène par des fonctions linéaires avant de chercher des modèles plus complexes. À la différence toutefois que le phénomène à modéliser dans une démarche d'AMCD n'est lui-même pas clairement défini, mais nous proposons une piste pour approcher ce problème à la section 9.2.

Nous avons présenté à la section 3.1 une méthode permettant de faire converger un groupe de décideurs vers un modèle de tri de type MR Sort unique. Les décideurs donnent au départ des exemples non nécessairement consensuels. La méthode leur fournit une manière de discuter de ces exemples tout en convergeant vers un modèle de tri partagé. La méthode adopte une démarche qui, dans le cadre d'une décision de groupe, est novatrice, en considérant deux types de paramètres séparément : les valeurs profils d'une part, les coalitions de cri-

tères de l'autre. Le consensus s'obtient donc en deux étapes. Concernant la première étape, nous avons défini des outils sous la forme de programmes mathématiques permettant de trouver des valeurs de profils consensuelles pour le groupe, avec des coalitions de critères non nécessairement partagées. La deuxième étape consiste à déterminer des valeurs de coalitions partagées, et une méthode existante peut être utilisée pour ce faire. Les outils proposés s'intègrent donc à l'existant de manière à former une méthodologie cohérente. La méthode proposée peut avoir pour effet de faciliter la convergence en divisant un problème complexe, l'obtention d'un modèle partagé, en deux sous-problèmes plus simples.

Nous avons présenté une application réelle illustrant l'usage de ces outils.

Nous avons également proposé un algorithme permettant de comparer deux modèles MR Sort sans veto. Cela est utile pour analyser la proximité entre deux décideurs, ou pour évaluer la performance d'une méthode d'inférence en comparant le modèle trouvé au modèle à trouver. L'algorithme calcule la probabilité que les deux modèles affectent une alternative différemment, étant donné une distribution de probabilité sur les performances des alternatives. L'algorithme permet de se calculer cette probabilité de manière exacte plutôt que de devoir l'estimer à partir d'un échantillonnage aléatoire. Il offre également la possibilité d'afficher une estimation en cours de calcul, estimation qui se précise au fur et à mesure de la progression du calcul.

Nous avons introduit une manière novatrice d'interroger le décideur d'une façon qui permet de prendre en compte ses hésitations, via l'expression de degrés de crédibilités, lorsqu'il fournit des exemples d'affectation. La méthode construit plusieurs modèles de préférence associés à des degrés de crédibilités, ces modèles pouvant être ordonnés par précision croissante et par degré de crédibilité décroissante. Cela est possible car notre méthode d'interrogation du décideur assure une certaine structure aux ensembles d'exemples fournis au départ et à leur association aux degrés de crédibilités. Les résultats de la méthode permettent au décideur de visualiser les compromis possibles entre la crédibilité et la précision des conclusions obtenues. Cette approche nous semble pouvoir être étendue dans un sens qui sera développé dans la section 9.2.

Nous avons proposé une méthode de choix de portefeuille. Elle intègre des préoccupations d'évaluation absolue, afin de s'assurer de la qualité intrinsèque des alternatives sélectionnées, et des préoccupations portant sur l'équilibre du portefeuille résultant. Les préférences concernant les alternatives individuelles sont définies de façon lâche, à l'aide d'exemples d'affectation, ce qui définit un ensemble de modèles de préférence compatibles et laisse de la latitude pour trouver un portefeuille qui respecte également des contraintes exprimées sur le groupe d'alternatives sélectionné.

Nous avons proposé un modèle de données en vue de faciliter l'échange de données de méthodes d'AMCD entre composants logiciels. Un schéma de données est actuellement utilisé pour normer l'encodage de telles données, mais notre proposition vise à résoudre un certain nombre d'inconvénients dans la façon dont le modèle actuel est implémenté. Au-delà de sa valeur pratique, cette contribution nous semble intéressante car elle permet de représenter les données utilisées par plusieurs méthodes d'AMCD dans un cadre général et concis.

Nous avons décrit les fonctionnalités implémentées dans deux bibliothèques libres et open source, JLP et J-MCDA. La première fournit un cadre logiciel pour faciliter la description de programmes mathématiques, la deuxième implémente les méthodes développées dans ce travail.

Au-delà des contributions méthodologiques, nous avons souhaité par cet ensemble de contributions de nature plus technique participer à la construction d'un environnement facilitant l'implémentation des méthodes proposées. Cela nous semble important, d'une part, pour encourager l'utilisation de ces méthodes qui restent parfois développées de manière théorique uniquement, et d'autre part pour permettre la reproduction, et donc la validation, des résultats publiés par les chercheurs.

Concernant ce dernier aspect, nous avons consacré beaucoup d'énergie à analyser les difficultés liées aux imprécisions numériques posées par l'implémentation des programmes mathématiques proposés dans ce document. C'est un travail qui est malheureusement parfois laissé de côté par la communauté scientifique, mais sans lequel les méthodes sont difficilement utilisables.

Au-delà des perspectives de recherche présentées ici et de celles développées en détail dans les chapitres concernés, nous souhaitons esquisser une perspective à plus long terme qui nous tient particulièrement à cœur.

9.2 Une ébauche de proposition pour la conception et la validation de modèles de préférence réfléchie

Nous avons brièvement abordé dans l'introduction de ce travail (Sections 1.4 et 1.10.1) la question de la validation des modèles de préférence. Cette section développe quelques réflexions concernant cette question. Cela requiert de définir plus précisément quels sont les objets que le modèle représente. Nous nous situons dans un cadre où le futur et le présent, capturés sous la forme des valeurs de performances des alternatives, sont connus avec certitude.

Nous avons indiqué (Section 1.1.6) qu'un modèle utile en aide à la décision

9 Conclusions et perspectives

devrait représenter des préférences « réfléchies », et non des préférences spontanées ou des décisions de terrain prises sans réflexions structurées. L'objet de cette section est de tenter de définir plus précisément cette notion de préférence réfléchie.

Le fait que les modèles utilisés en AMCD représentent autre chose que des actes observables interdit une validation simple consistant à comparer les prédictions du modèle aux actions effectuées par le décideur. Dans le champ de la recherche opérationnelle, il est souvent admis que les modèles utilisés s'écartent du schéma classique de validation d'un modèle par sa capacité prédictive (Ackoff, 1979; Landry *et al.*, 1983). Le problème de validation des modélisations utilisées en recherche opérationnelle est plus vaste que la seule question de la modélisation des préférences qui nous intéresse dans ce travail, car une difficulté importante de la modélisation d'un problème opérationnel tient à la difficulté de s'assurer de l'adéquation du modèle à la réalité et de son utilité et son utilisabilité pour le décideur (Landry *et al.*, 1983). La modélisation doit aussi être capable de représenter des incertitudes, des avis divergeants dans un groupe, etc. Nous n'aborderons pas ces questions. Cependant, nous pensons qu'une partie de la difficulté de validation des modèles utilisés vient de la difficulté de la gestion des préférences, et de la particularité du champ, par rapport aux sciences naturelles telles que la physique, émanant de la prise en compte d'aspects subjectifs.

Le point de vue plus spécifique de l'AMCD concernant la validation de modélisation des préférences est discuté par Genard et Pirlot (2002). Ils font remarquer que la question du statut épistémologique des modèles utilisés en recherche opérationnelle et en AMCD n'a pas encore reçue de réponse claire. Ils proposent de valider la procédure d'AMCD plutôt que le modèle, et indiquent : « Notre sentiment est que, bien que les modèles et recommandations utilisés en aide à la décision (et en recherche opérationnelle) ne puissent prétendre au statut de "vrai" ou de "bon", la façon dont ils sont agencés, dans la perspective constructiviste, nous semble reproduire les conditions procédurales qui entraîneraient leur validité s'ils étaient implémentés après que toutes les personnes intéressées par la décision aient pris part au processus. » (Genard et Pirlot, 2002, p. 5, traduit par nos soins). Dans la mesure où l'analyste ne s'adresse pas en pratique à toutes les parties potentiellement intéressées par la décision, les auteurs indiquent que « les modèles et recommandations issus du processus d'aide à la décision ne peuvent, en toute probabilité, pas être a priori reconnus comme dotés d'une validité "externe" » (Genard et Pirlot, 2002, p. 12). La validité des modèles et résultats est déterminée, pour le point de vue constructiviste en général, par leur acceptation par le décideur et par l'interaction avec

l'analyste¹.

9.2.1 Modéliser les préférences réfléchies

Nous pensons qu'une piste de recherche intéressante consiste à inscrire une partie de l'activité d'AMCD dans une perspective réaliste classique où le modèle décrit un aspect de la réalité et peut y être confronté par ses prédictions. Pour ce faire, il faudrait mieux définir les types de préférences que le modèle cherche à capturer.

Nous avons parlé des préférences sous la forme d'actes, par exemple d'achats : c'est le champ de la modélisation descriptive des préférences. Les constructivistes s'intéressent à ces préférences qui sont construites par le processus d'aide à la décision. Mais on pourrait adopter un autre axe d'analyse. Nous proposons de plutôt tenter de modéliser des préférences réfléchies, définies comme suit : les préférences réfléchies sont les affirmations de préférences qui résistent aux contre-arguments.

Le décideur dont les préférences sont analysées pourrait être confronté à plusieurs arguments supportant des conclusions opposées. Par exemple, certains arguments pourraient être des arguments de cohérence : le décideur a indiqué qu'il préférerait cette option à celle-là, donc pour être cohérent selon telle classe de modèles, il devrait aussi avoir telle préférence. Des situations telles que celle mentionnée page 47 pourraient se prêter à de tels tests. La préférence réfléchie serait celle que le décideur exprime après avoir réfléchi à tous les arguments. Le modèle de préférence aurait pour tâche de représenter ces préférences réfléchies : il devrait donc être capable de prédire ce type de réponses. Cela fournit une manière de le falsifier, au sens de Popper (1959).

1. Voir aussi la description de Bouyssou (1984, p. 14) : « La pratique des études d'aide à la décision, qu'elles soient fondées ou non sur des considérations axiomatiques, montre qu'une des qualités essentielles de l'étude doit être de se faire accepter, c'est-à-dire de bien s'insérer dans le processus de décision dont elle n'est qu'une étape (...). En un mot, elle doit convaincre les personnes à qui elle s'adresse mais aussi l'"homme d'étude" [l'analyste, selon notre terminologie] qui la met en œuvre (...). On pourra ici nous objecter que nous passons d'une attitude fort rationaliste à sa négation même. Si nous définissons en effet un "bon" modèle d'aide à la décision (c'est-à-dire un modèle qui a passé le test de sa validité avec succès), comme un modèle qui emporte la double conviction de l'homme d'étude et du décideur, nous rendons la définition contingente à la personnalité des personnes en présence ainsi qu'à la nature du problème étudié. Ce point de vue, s'il peut paraître choquant, nous semble cependant le seul utile en Recherche Opérationnelle. Il implique, certes, un certain pragmatisme mais il nous semble souhaitable d'introduire explicitement cette notion dès lors que, et c'est notre cas, on considère qu'il n'existe pas de solution unique pour résoudre un problème donné. » Nous renvoyons également le lecteur à la citation de Bernard Roy indiquée page 22.

Notons que cette idée d'observer la stabilité des préférences face aux arguments a été utilisée, bien que dans un autre contexte, par Stanovich et West (1999). Dans cette étude, les auteurs posent des questions de décision, sous la forme de choix. Il s'agit par exemple de décider de coopérer ou non, dans un contexte de théorie des jeux. Dans une deuxième phase, les expérimentateurs soumettent des arguments aux sujets, en faveur de l'option qu'ils ont choisie ou en faveur de l'autre option. Ils étudient ensuite la propension des individus à modifier leur choix. L'étude ne concerne cependant pas explicitement une modélisation de compromis entre plusieurs critères comme on souhaite la pratiquer en AMCD. Une autre différence est que la perspective des auteurs est de tester le principe de compréhension/acceptation de Slovic et Tversky (1974), ils considèrent donc les options de façon asymétrique : l'une est le choix approprié d'après une certaine norme. Mais cette idée pourrait être appliquée pour tester les modèles de préférences réfléchies d'AMCD, sans nécessairement postuler de norme.

Un article récent (Greco *et al.*, 2013) propose d'expliquer des conclusions (éventuellement robustes) en représentant le modèle de préférence par un ensemble de règles, au sens exposé à la section 2.1.2. Bien que l'article n'aborde pas les questions de validation qui nous intéressent ici, les règles pourraient être vues comme des arguments venant soutenir les conclusions. Cet article pourrait donc constituer un point d'appui utile pour notre proposition.

Nous parlerons ci-dessous d'une autre étude intéressante liée à l'étude du lien entre les préférences et les arguments après avoir exposé plus précisément le type de préférence qui nous intéresse.

9.2.2 Deux types de préférences réfléchies

Précisons le type de préférence que nous pensons que le modèle devrait représenter. Comme le font remarquer les constructivistes (voir Section 1.4.1), les préférences peuvent être construites durant le processus d'AMCD. Il est selon nous important de distinguer deux types de constructions.

Il est possible que certaines préférences réfléchies s'établissent durant l'interaction entre l'analyste et le décideur, dans le sens suivant : l'affirmation de préférence que le décideur indiquera comme étant sa préférence « tout compte fait », c'est-à-dire après avoir réfléchi à tous les arguments, dépend du déroulement du processus d'aide à la décision. Par exemple, l'ordre des questions, ou l'ordre par lequel les arguments sont présentés, pourrait modifier durablement les préférences réfléchies du décideur.

À l'inverse (mais sans contradiction avec la proposition précédente), il est possible que certaines préférences réfléchies ne dépendent pas du déroulement du processus. Cela ne signifie pas qu'elles seront identiques aux préférences

spontanées : après réflexion, la préférence peut être différente de la préférence exprimée avant confrontation du décideur aux arguments. Mais il est possible que cette transformation de préférence spontanée en préférence réfléchie ne dépende pas de la façon d'interroger le décideur lors de l'établissement du modèle de préférence. C'est ce deuxième type de préférence que nous suggérons de modéliser, de manière à ce que le modèle donne des conclusions indépendantes du déroulement du processus visant à collecter les données de préférence.

Une difficulté importante est liée à la représentation des arguments sur lesquels un modèle de préférence réfléchie tel que nous le proposons devrait s'appuyer. Le champ de la théorie de l'argumentation (Rahwan et Simari, 2009) propose des cadres formels permettant de manipuler des arguments. Des travaux récents s'intéressent au rapprochement entre l'AMCD et la théorie de l'argumentation (Amgoud *et al.*, 2005 ; Caminada et Pigozzi, 2011 ; Ouerdane, 2009 ; Ouerdane *et al.*, 2010). C'est là probablement un terrain de recherche très prometteur.

9.2.3 Les préférences réfléchies pourraient être incomplètes

Nous postulons qu'il devrait être possible de trouver des circonstances où le décideur est prêt à accepter deux argumentations menant à deux conclusions contradictoires, sans être capable de déterminer facilement laquelle des deux il préfère, même après avoir examiné tous les arguments en faveur et en défaveur de chaque conclusion. C'est ce que nous appelons des préférences réfléchies incomplètes.

Disposer d'une théorie permettant de rendre compte, voire de prédire, de telles situations serait précieux dans un contexte de décision de groupe. Exploiter ces zones où les préférences réfléchies d'un individu ne sont pas déterminées pourrait faciliter la recherche de consensus.

Sur le plan de la modélisation, cela signifie qu'un modèle de préférence réfléchie pourrait être un modèle imprécis, au sens de la section 1.5. Un tel modèle a cependant des liens étroits avec l'approche robuste (Section 1.6) qui utilise également des fonctions imprécises. Il est possible que dans certains cas un modèle imprécis qui donnerait des conclusions calculées par l'approche robuste puisse être construit. Une telle construction est esquissée à la section 2.3.2. Une piste de recherche future est d'explorer plus avant les liens entre les modèles imprécis et l'approche robuste. Là encore une des difficultés consiste à formaliser les arguments qui pourraient être construits sur base d'une fonction de préférence robuste.

Dans une étude particulièrement intéressante pour notre propos, Lichtenstein et Slovic (2006c) ont examiné le phénomène du renversement de préférence de sujets lors de comparaisons entre deux loteries. Il s'agit là d'une étude

relativement classique : le sujet est soumis à deux présentations différentes qui l'amènent à se concentrer sur le montant gagné dans un cas, et sur le risque dans l'autre, sans qu'il s'en rende compte. Cela le conduit à affirmer des préférences contradictoires. Cette étude est néanmoins particulièrement intéressante car les auteurs ont interrogé les sujets après coup : ils les ont confrontés à certaines de leurs affirmations contradictoires, et leur ont soumis des arguments dans le but de modifier leur préférence. L'étude citée (Lichtenstein et Slovic, 2006c) est une nouvelle édition d'une étude précédente (Lichtenstein et Slovic, 1971), mais cette édition comprend en annexe ce dialogue entre une auteure et un sujet, dont nous recommandons la lecture. Les auteurs indiquent que parmi les onze sujets interrogés, six ont modifié leur préférence sans que beaucoup d'efforts de persuasion soient nécessaires, trois ont modifié leur préférence après plus d'insistance, et deux ont refusé de modifier leur réponse. Dans cette étude, le but des auteurs n'était pas de révéler l'existence de préférences réfléchies incomplètes et ils demandaient donc aux sujets de choisir une seule façon de voir. Mais il nous paraît possible que certains de ces individus auraient saisi l'opportunité, si on la leur avait présentée, de déclarer que les deux façons de voir leur semblaient correctes : leur façon originale de voir, et la façon de voir normative présentée avec force arguments par les auteurs.

Les études décrivant des préférences s'approchant de notre concept de préférences réfléchies sont peu courantes. Plus rares encore sont les études laissant la possibilité au décideur d'indiquer une hésitation. Certaines études toutefois font exception : dans ces expériences, le sujet peut reporter sa décision ou payer pour avoir plus d'informations (Tversky et Shafir, 1992 ; Luce, 1998). Dans les expériences menées par Deparis (2012) (d'où les deux références précédentes sont tirées), le sujet peut faire savoir directement son hésitation (Deparis *et al.*, 2012). Mais ces études ne font pas référence à l'argumentation éventuellement sous-jacente aux choix du sujet.

Nous avons présenté la question de la complétude des préférences réfléchies de façon binaire : pour une conclusion donnée, elles sont soit déterminées, soit indéterminées dans le cas où plusieurs arguments contradictoires ne pourraient être départagés. Mais une représentation plus nuancée de ce genre de situations serait sans-doute utile. De manière similaire à l'approche présentée par Greco *et al.* (2010) (Section 2.3.2) et l'approche détaillée au chapitre 4, la représentation appropriée pourrait être un modèle donnant des conclusions associées à un niveau de crédibilité. Ainsi, dans certaines situations, le décideur trouverait aussi crédible deux argumentations conduisant à des conclusions opposées et serait incapable de trancher. Dans d'autres, il trouverait une argumentation crédible, mais l'autre encore plus crédible, et cela le conduirait à affirmer une préférence. Des travaux existants en théorie de l'argumentation visent à com-

parer les forces de différents arguments, ils pourraient donc peut-être constituer une source d'inspiration utile (Dubois *et al.*, 2008).

Précisons si cela est nécessaire que nous ne prétendons pas contourner la loi de Hume (2003), selon laquelle un ensemble d'affirmations uniquement factuelles ne peut impliquer logiquement une recommandation : l'approche que nous proposons ici, s'inscrivant dans une épistémologie réaliste, ne peut mener à elle seule (c'est-à-dire sans lui adjoindre de considérations morales) à des recommandations valides. Notre proposition ne concerne donc qu'un aspect de l'activité de l'AMCD.

Par ailleurs, beaucoup de travail est encore nécessaire avant d'aboutir à une formulation précise des concepts présentés ici et cet exposé n'est, répétons-le, qu'une très modeste ébauche. Nous pensons pourtant que cette approche pourra contribuer, à côté des approches existantes, au champ de recherche visant à modéliser la subjectivité des individus.

Ainsi qu'exposé brièvement à la section 1.10.1, nous pensons que le champ de la modélisation des préférences peut contribuer, bien que modestement, à une des raisons d'être de la science : doter le genre humain d'analyses permettant de mieux décider ensemble où l'on souhaite aller. Et il y a certainement de quoi s'améliorer en la matière...

Bibliographie

Bibliographie

- M. ABDELLAOUI, C. BARRIOS et P. P. WAKKER : Reconciling introspective utility with revealed preference: Experimental arguments based on prospect theory. *Journal of Econometrics*, 138(1):356–378, mai 2007. ISSN 0304-4076. URL <http://www.sciencedirect.com/science/article/pii/S0304407606000984>.
- R. L. ACKOFF : The future of operational research is past. *The Journal of the Operational Research Society*, 30(2):93–104, 1979. ISSN 01605682.
- L. AMGOUD, J.-F. BONNEFON et H. PRADE : An argumentation-based approach to multiple criteria decision. In L. GODO, éd. : *Symbolic and Quantitative Approaches to Reasoning with Uncertainty*, vol. 3571 de *Lecture Notes in Computer Science*, p. 269–280. Springer Berlin Heidelberg, Berlin, Heidelberg, 2005. ISBN 978-3-540-27326-4, 978-3-540-31888-0. URL <http://www.springerlink.com/content/0qp68frcf2737md1/>.
- T. AVEN : *Foundations of Risk Analysis*. John Wiley & Sons, 2003. ISBN 9781119945789.
- T. AVEN : *Risk Analysis: Assessing Uncertainties Beyond Expected Values and Probabilities*. John Wiley & Sons, 2008. ISBN 9780470517369.
- T. AVEN : *Misconceptions of Risk*. John Wiley & Sons, 2010. ISBN 9781119964285.
- D. BELL, H. RAIFFA et A. TVERSKY, éd. *Decision making: descriptive, normative, and prescriptive interactions*. Cambridge university press, Cambridge, 1988.
- V. BELTON et T. STEWART : *Multiple Criteria Decision Analysis: An Integrated Approach*. Kluwer Academic, Dordrecht, 2002.
- J. BESHEARS, J. J. CHOI, D. LAIBSON et B. C. MADRIAN : How are preferences revealed? *Journal of Public Economics*, 92(8-9):1787–1794, août 2008. ISSN 00472727. URL <https://www.sciencedirect.com/science/article/pii/S0047272708000728>.

Bibliographie

- S. BIGARET et P. MEYER : diviz: an MCDA workflow design, execution and sharing tool. *Intelligent Decision Technologies Journal*, 2012. To appear.
- G. BOUS, P. FORTEMPS, F. GLINEUR et M. PIRLOT : ACUTA: a novel method for eliciting additive value functions on the basis of holistic preference statements. *European Journal of Operational Research*, 206(2):435–444, oct. 2010. ISSN 0377-2217. URL <http://www.sciencedirect.com/science/article/pii/S0377221710001827>.
- D. BOUYSSOU : *Approches descriptives et constructives d'aide à la décision : Fondements et comparaison*. Thèse de 3ème cycle, Univ. Paris Dauphine, 1984.
- D. BOUYSSOU et T. MARCHANT : Ordered categories and additive conjoint measurement on connected sets. *Journal of Mathematical Psychology*, 53(2):92–105, avr. 2009. ISSN 0022-2496. URL <http://www.sciencedirect.com/science/article/pii/S0022249608001156>.
- D. BOUYSSOU, D. DUBOIS, M. PIRLOT et H. PRADE : *Concepts et méthodes pour l'aide à la décision, volume 3, analyse multicritère*. Hermès, 2006a.
- D. BOUYSSOU et T. MARCHANT : An axiomatic approach to non-compensatory sorting methods in MCDM, I : The case of two categories. *European Journal of Operational Research*, 178(1):217–245, 2007a. ISSN 0377-2217. URL <http://www.sciencedirect.com/science/article/B6VCT-4JVSWMR-1/2/e304a3c0af4c11e5d22f7b1dd1587dd7>.
- D. BOUYSSOU et T. MARCHANT : An axiomatic approach to noncompensatory sorting methods in MCDM, II: more than two categories. *European Journal of Operational Research*, 178(1):246–276, avr. 2007b. ISSN 0377-2217. URL <http://www.sciencedirect.com/science/article/pii/S0377221706000580>.
- D. BOUYSSOU et T. MARCHANT : Additive and decomposable conjoint measurement with ordered categories. *Cahier du LAMSADE*, (275), 2008. URL <http://basepub.dauphine.fr/xmlui/handle/123456789/5096>.
- D. BOUYSSOU et T. MARCHANT : Additive conjoint measurement with ordered categories. *European Journal of Operational Research*, 203(1):195 – 204, 2010. ISSN 0377-2217. URL <http://www.sciencedirect.com/science/article/pii/S0377221709005189>.

- D. BOUYSSOU et T. MARCHANT : Biorders with frontier. *Order*, 28(1):53–87, 2011. ISSN 0167-8094. URL <http://www.springerlink.com/content/82t674526h826k04/abstract/>.
- D. BOUYSSOU, T. MARCHANT, M. PIRLOT, A. TSOUKIÀS et P. VINCKE : *Evaluation and decision models with multiple criteria: Stepping stones for the analyst*. International Series in Operations Research and Management Science, Volume 86. Springer, Boston, 1st édn, 2006b. ISBN 0-387-31098-3. URL <http://www.springer.com/sgw/cda/frontpage/0,11855,4-40521-22-116132747-0,00.html>.
- J. P. BRANS : L'ingénierie de la décision; élaboration d'instruments d'aide à la décision. la méthode PROMETHEE. In R. NADEAU et M. LANDRY, édés : *L'aide à la décision: Nature, Instruments et Perspectives d'Avenir*, p. 183–213, Québec, Canada, 1982. Presses de l'Université Laval.
- J. BRANS et B. MARESCHAL : PROMETHEE methods. In J. FIGUEIRA, S. GRECO et M. EHRGOTT, édés : *Multiple Criteria Decision Analysis: State of the Art Surveys*, p. 163–196. Springer Verlag, Boston, Dordrecht, London, 2005. URL <http://www.springeronline.com/sgw/cda/frontpage/0,11855,5-165-22-34954528-0,00.html>.
- J. BRANS et P. VINCKE : A preference ranking organization method. *Management Science*, 31(6):647–656, 1985.
- O. CAILLOUX : ELECTRE and PROMETHEE MCDA methods as reusable software components. In *Proceedings of the 25th Mini-EURO Conference on Uncertainty and Robustness in Planning and Decision Making (URPDM 2010)*, Coimbra, Portugal, avr. 2010. University of Coimbra, Portugal. ISBN 978-989-95055-3-7.
- O. CAILLOUX : Dealing with numerical imprecision in mathematical programs for electre tri models disaggregation. Rap. tech. 2012-02, Laboratoire Génie Industriel, École Centrale Paris, mars 2012. URL <http://www.lgi.ecp.fr/Biblio/PDF/CR-LGI-2012-02.pdf>. Cahiers de recherche 2012-02.
- O. CAILLOUX, B. MAYAG, P. MEYER et V. MOUSSEAU : An operational tool to build a multicriteria territorial risk scale with multiple experts. *Reliability Engineering & System Safety*, juil. 2012a. Submitted.
- O. CAILLOUX, P. MEYER et V. MOUSSEAU : Eliciting ELECTRE TRI category limits for a group of decision makers. *European Journal of Operational Research*, 223(1):133–140, 2012b. URL <http://www.sciencedirect.com/science/article/pii/S0377221712003906?v=s5>.

Bibliographie

- O. CAILLOUX et V. MOUSSEAU : Parameterize a territorial risk evaluation scale using multiple experts knowledge through risk assessment examples. In C. BÉRENGUER, A. GRALL et C. GUEDES SOARES, édés : *Advances in Safety, Reliability and Risk Management*, p. 2331–2339, Troyes, France, sept. 2011. Taylor and Francis Group, London. ISBN 978-0-415-68379-1. URL <http://www.esrel2011.com/>.
- O. CAILLOUX, T. TERVONEN, B. VERHAEGEN et F. PICALAUSA : A data model for algorithmic multiple criteria decision analysis. *Decision Sciences*, déc. 2012c. ISSN 0011-7315. Submitted.
- G. CALVÈS : *La discrimination positive*. Presses Universitaires de France - PUF, 3 édén, oct. 2010. ISBN 2130583318.
- M. CAMINADA et G. PIGOZZI : On judgment aggregation in abstract argumentation. *Autonomous Agents and Multi-Agent Systems*, 22(1):64–102, 2011. ISSN 1387-2532. URL <http://www.springerlink.com/content/962044h73u063702/>.
- S. DAMART, L. DIAS et V. MOUSSEAU : Supporting groups in sorting decisions: Methodology and use of a multi-criteria aggregation/disaggregation DSS. *Decision Support Systems*, 43(4):1464–1475, août 2007.
- D. DENNETT : *De Beaux Reves*. Gallimard Education, fév. 2012. ISBN 2070443485.
- D. C. DENNETT : *Freedom Evolves*. Penguin, advance reading copy (ARC) édén, 2002.
- S. DEPARIS : *Etude de l'Effet du Conflit Multicritère sur l'Expression des Préférences : une Approche Empirique*. Thèse de doctorat, École Centrale Paris, Châtenay-Malabry, France, juin 2012.
- S. DEPARIS, V. MOUSSEAU, M. ÖZTÜRK, C. PALLIER et C. HURON : When conflict induces the expression of incomplete preferences. *European Journal of Operational Research*, 221(3):593–602, sept. 2012. ISSN 03772217. URL <http://www.sciencedirect.com/science/article/pii/S0377221712002512>.
- J. DEVAUD, G. GROUSSAUD et E. JACQUET-LAGRÈZE : UTADIS: une méthode de construction de fonctions d'utilité additives rendant compte de jugements globaux. In *European working group on MCDA, Bochum, Germany*, 1980.

- L. DIAS et J. CLÍMACO : Dealing with imprecise information in group multicriteria decisions: a methodology and a GDSS architecture. *European Journal of Operational Research*, 160(2):291–307, jan. 2005. ISSN 0377-2217. URL <http://www.sciencedirect.com/science/article/pii/S0377221703006131>.
- L. DIAS et V. MOUSSEAU : IRIS: a DSS for multiple criteria sorting problems. *Journal of Multi-Criteria Decision Analysis*, 12:285–298, 2003. URL <http://www3.interscience.wiley.com/cgi-bin/jhome/5725>.
- L. DIAS, V. MOUSSEAU, J. FIGUEIRA et J. CLÍMACO : An aggregation/disaggregation approach to obtain robust conclusions with ELECTRE TRI. *European Journal of Operational Research*, 138(2):332–348, avr. 2002. URL <http://www.sciencedirect.com/science/article/pii/S0377221701002508>.
- M. DOUMPOS, Y. MARINAKIS, M. MARINAKI et C. ZOPOUNIDIS : An evolutionary approach to construction of outranking models for multicriteria classification: The case of the ELECTRE TRI method. *European Journal of Operational Research*, 199(2):496–505, déc. 2009. ISSN 0377-2217. URL <http://www.sciencedirect.com/science/article/pii/S0377221708010163>.
- M. DOUMPOS et C. ZOPOUNIDIS : *Multicriteria Decision Aid Classification Methods*. Kluwer Academic Publishers, Dordrecht, 2002.
- D. DUBOIS, H. FARGIER et J. BONNEFON : On the qualitative comparison of decisions having positive and negative features. *Journal of Artificial Intelligence Research*, 32:385–417, 2008. ISSN 1076 - 9757.
- P. H. FARQUHAR et V. R. RAO : A balance model for evaluating subsets of multiattributed items. *Management Science*, 22(5):528–539, 1976. URL <http://mansci.journal.informs.org/cgi/content/abstract/22/5/528>.
- J. FEDOROWICZ et G. B. WILLIAMS : Representing modeling knowledge in an intelligent decision support system. *Decision Support Systems*, 2(1):3–14, mars 1986. ISSN 0167-9236. URL <http://www.sciencedirect.com/science/article/pii/0167923686901168>.
- J. FIGUEIRA, S. GRECO et M. EHRGOTT : *Multiple Criteria Decision Analysis: State of the Art Surveys*. Springer Verlag, Boston, Dordrecht, London, 2005a. URL <http://www.springeronline.com/sgw/cda/frontpage/0,11855,5-165-22-34954528-0,00.html>.
- J. FIGUEIRA, V. MOUSSEAU et B. ROY : ELECTRE methods. In J. FIGUEIRA, S. GRECO et M. EHRGOTT, édés : *Multiple Criteria Decision Analysis: State*

Bibliographie

- of the Art Surveys*, p. 133–162. Springer Verlag, Boston, Dordrecht, London, 2005b. URL <http://www.springeronline.com/sgw/cda/frontpage/0,11855,5-165-22-34954528-0,00.html>.
- B. FISCHHOFF : Value elicitation: Is there anything there? *American psychologist*, 46:835–847, 1991.
- E. H. FORMAN et S. I. GASS : The analytic hierarchy Process—An exposition. *Operations Research*, 49(4):469–486, juil. 2001. ISSN 0030-364X, 1526-5463. URL <http://or.journal.informs.org/content/49/4/469>.
- R. FOURER, H. I. GASSMANN, J. MA et R. K. MARTIN : An XML-based schema for stochastic programs. *Annals of Operations Research*, 166(1):313–337, fév. 2009. ISSN 0254-5330, 1572-9338. URL <http://www.springerlink.com/content/d0t2x528q7p426p2/>.
- R. FOURER, J. MA et K. MARTIN : Optimization services: A framework for distributed optimization. *Operations Research*, 58(6):1624–1636, nov. 2010a. ISSN 0030-364X, 1526-5463. URL <http://or.journal.informs.org/content/58/6/1624>.
- R. FOURER, J. MA et K. MARTIN : OSiL: an instance language for optimization. *Computational Optimization and Applications*, 45(1):181–203, jan. 2010b. ISSN 0926-6003. 10.1007/s10589-008-9169-6.
- I. FUJIMURA : La féminisation des noms de métiers et des titres dans la presse française (1988-2001). *Mots. Les langages du politique*, 2005(78):37–52, juil. 2005. ISSN 0243-6450. URL <http://mots.revues.org/355>.
- J. FÜRNKRANZ et E. HÜLLERMEIER : Preference learning: An introduction. *In Preference Learning*, p. 1 – 17. Springer, 2011. ISBN 978-3-642-14124-9. URL <http://www.springerlink.com/content/u32t275070q2751w/>.
- L. GAUTHIER et T. NÉEL : SAGE: an object-oriented framework for the construction of farm decision support systems. *Computers and Electronics in Agriculture*, 16(1):1–20, déc. 1996. ISSN 0168-1699. URL <http://www.sciencedirect.com/science/article/pii/S016816999600018X>.
- J.-L. GENARD et M. PIRLOT : Multi-criteria decision aid in a philosophical perspective. *In D. BOUYSSOU, E. JACQUET-LAGRÈZE, P. PERNY, R. SLOWINSKI et D. VANDERPOOTEN, édés : Aiding Decisions with Multiple Criteria. Essays in Honor of Bernard Roy*, vol. 44 de *International Series in Operations Research & Management Science*, p. 89–117. Springer, 2002.

- E. GEORGOPOULOU, Y. SARAFIDIS et D. DIAKOULAKI : Design and implementation of a group DSS for sustaining renewable energies exploitation. *European Journal of Operational Research*, 109(2):483–500, sept. 1998. ISSN 0377-2217. URL <http://www.sciencedirect.com/science/article/pii/S0377221798000721>.
- K. GOLABI, C. W. KIRKWOOD et A. SICHERMAN : Selecting a portfolio of solar energy projects using multiattribute preference theory. *Management Science*, 27(2):174–189, 1981. URL <http://mansci.journal.informs.org/cgi/content/abstract/27/2/174>.
- S. GRECO, B. MATARAZZO et R. SŁOWIŃSKI : Conjoint measurement and rough set approach for multicriteria sorting problems in presence of ordinal criteria. In A. COLORNI, M. PARUCCINI et B. ROY, édés : *A-MCD-A, Aide Multicritère à la Décision/Multiple Criteria Decision Aid*, p. 117–144. European Commission, Joint Research Centre, EUR 19808 EN, Ispra, 2001a.
- S. GRECO, B. MATARAZZO et R. SŁOWIŃSKI : Rough sets theory for multicriteria decision analysis. *European Journal of Operational Research*, 129(1): 1–47, 2001b. URL <http://www.sciencedirect.com/science/article/pii/S0377221700001673>.
- S. GRECO, B. MATARAZZO et R. SŁOWIŃSKI : Rough sets methodology for sorting problems in presence of multiple attributes and criteria. *European Journal of Operational Research*, 138(2):247–259, 2002.
- S. GRECO, B. MATARAZZO et R. SŁOWIŃSKI : Decision rule approach. In J. FIGUEIRA, S. GRECO et M. EHRGOTT, édés : *Multiple Criteria Decision Analysis: State of the Art Surveys*, p. 507–562. Springer Verlag, Boston, Dordrecht, London, 2005. URL <http://www.springeronline.com/sgw/cda/frontpage/0,11855,5-165-22-34954528-0,00.html>.
- S. GRECO, V. MOUSSEAU et R. SŁOWIŃSKI : Ordinal regression revisited: multiple criteria ranking using a set of additive value functions. *European Journal of Operational Research*, 191(2):415–435, déc. 2008. URL <http://www.sciencedirect.com/science/article/pii/S0377221707008752>.
- S. GRECO, V. MOUSSEAU et R. SŁOWIŃSKI : The possible and the necessary for multiple criteria group decision. *Algorithmic Decision Theory*, 5783:203–214, 2009. ISSN 0302-9743. URL <http://www.springer.com/computer/theoretical+computer+science/book/978-3-642-04427-4>.
- S. GRECO, R. SŁOWIŃSKI et P. ZIELNIEWICZ : Putting dominance-based rough set approach and robust ordinal regression together. *Decision Support*

Bibliographie

- Systems*, 2013. ISSN 0167-9236. URL <http://www.sciencedirect.com/science/article/pii/S0167923612002485?v=s5>. In press.
- S. GRECO, M. KADZIŃSKI, V. MOUSSEAU et R. SŁOWIŃSKI : Robust ordinal regression for multiple criteria group decision: UTAGMS-GROUP and UTADISGMS-GROUP. *Decision Support Systems*, 52(3):549–561, fév. 2012. ISSN 01679236. URL <http://linkinghub.elsevier.com/retrieve/pii/S0167923611001680>.
- S. GRECO, M. KADZIŃSKI et R. SŁOWIŃSKI : Selection of a representative value function in robust multiple criteria sorting. *Computers & Operations Research*, 38(11):1620–1637, nov. 2011. ISSN 03050548. URL <http://linkinghub.elsevier.com/retrieve/pii/S0305054811000438>.
- S. GRECO, V. MOUSSEAU et R. SŁOWIŃSKI : Multiple criteria sorting with a set of additive value functions. *European Journal of Operational Research*, 207(3):1455–1470, déc. 2010. ISSN 0377-2217. URL <http://www.sciencedirect.com/science/article/pii/S0377221710003772>.
- A. GUAZZELLI, M. ZELLER, W. CHEN et G. WILLIAMS : PMML: an open standard for sharing models. *The R Journal*, 1:60–65, mai 2009.
- D. HOCHBAUM et A. LEVIN : Methodologies and algorithms for group-rankings decision. *Management Science*, 52(9):1394–1408, 2006.
- I. B. HONG et D. R. VOGEL : Data and model management in a generalized MCDM-DSS. *Decision Sciences*, 22(1):1–25, jan. 1991. ISSN 0011-7315. URL <http://onlinelibrary.wiley.com/doi/10.1111/j.1540-5915.1991.tb01258.x/abstract>.
- D. HUME : *A Treatise of Human Nature*. Courier Dover Publications, nov. 2003. ISBN 9780486432502.
- D. C. INCE, L. HATTON et J. GRAHAM-CUMMING : The case for open computer programs. *Nature*, 482(7386):485–488, fév. 2012. ISSN 0028-0836. URL <http://www.nature.com/nature/journal/v482/n7386/abs/nature10836.html>.
- E. JACQUET-LAGRÈZE et Y. SISKOS : Assessing a set of additive utility functions for multicriteria decision making: the UTA method. *European Journal of Operational Research*, 10:151–164, 1982.
- A. JIMÉNEZ, S. RÍOS-INSUA et A. MATEOS : A generic multi-attribute analysis system. *Computers & Operations Research*, 33(4):1081–1101, avr. 2006. ISSN

- 0305-0548. URL <http://www.sciencedirect.com/science/article/pii/S0305054804002266>.
- K. JÁRMAI : Single- and multicriteria optimization as a tool of decision support system. *Computers in Industry*, 11(3):249–266, jan. 1989. ISSN 0166-3615. URL <http://www.sciencedirect.com/science/article/pii/0166361589900067>.
- D. KAHNEMAN et A. TVERSKY, édés. *Choices, Values, and Frames*. Cambridge University Press, 2000. ISBN 9780521627498.
- R. KEENEY et H. RAIFFA : *Decisions with multiple objectives: Preferences and value tradeoffs*. J. Wiley, New York, 1976.
- T. KOCH, T. ACHTERBERG, E. ANDERSEN, O. BASTERT, T. BERTHOLD, R. E. BIXBY, E. DANNA, G. GAMRATH, A. M. GLEIXNER, S. HEINZ, A. LODI, H. MITTELMANN, T. RALPHS, D. SALVAGNIN, D. E. STEFFY et K. WOLTER : MIPLIB 2010. *Mathematical Programming Computation*, 3(2):103–163, 2011. URL <http://mpc.zib.de/index.php/MPC/article/view/56/28>.
- R. LAHDELMA, J. HOKKANEN et P. SALMINEN : SMAA - stochastic multiobjective acceptability analysis. *European Journal of Operational Research*, 106(1):137–143, 1998.
- R. LAHDELMA et P. SALMINEN : SMAA-2: stochastic multicriteria acceptability analysis for group decision making. *Operations Research*, 49(3):444–454, mai 2001. URL <http://or.journal.informs.org/content/49/3/444.abstract>.
- M. LANDRY, J. MALOUIN et M. ORAL : Model validation in operations research. *European Journal of Operational Research*, 14:207–220, 1983. URL <http://www.sciencedirect.com/science/article/pii/0377221783902576>.
- J. LE CARDINAL, V. MOUSSEAU et J. ZHENG : Multiple criteria sorting: An application to student selection. In A. SALO, J. KEISLER et A. MORTON, édés : *Portfolio Decision Analysis*. Springer-Verlag New York Inc., juil. 2011. ISBN 1441999426.
- A. LEROY, V. MOUSSEAU et M. PIRLOT : Learning the parameters of a multiple criteria sorting method. In R. BRAFMAN, F. ROBERTS et A. TSOUKIÀS, édés : *Algorithmic Decision Theory*, vol. 6992 de *Lecture Notes in Computer*

Bibliographie

- Science*, p. 219–233. Springer Berlin / Heidelberg, 2011. ISBN 978-3-642-24872-6. URL dx.doi.org/10.1007/978-3-642-24873-3_17.
- S. LICHTENSTEIN et P. SLOVIC : Reversals of preference between bids and choices in gambling decisions. *Journal of Experimental Psychology*, 89:46–55, 1971.
- S. LICHTENSTEIN et P. SLOVIC : *The Construction of Preference*. Cambridge University Press, 1^{er} éd., 2006a. ISBN 0521834287.
- S. LICHTENSTEIN et P. SLOVIC : The construction of preference: An overview. *In The Construction of Preference*, p. 1 – 40. Cambridge University Press, 2006b. ISBN 0521834287.
- S. LICHTENSTEIN et P. SLOVIC : Reversals of preferences between bids and choices gambling decisions. *In S. LICHTENSTEIN et P. SLOVIC, édés : The Construction of Preference*, p. 52–68. Cambridge University Press, 1^{er} éd., 2006c. ISBN 0521834287.
- J. LIESIÖ, P. MILD et A. SALO : Preference programming for robust portfolio modeling and project selection. *European Journal of Operational Research*, 181(3):1488 – 1505, sept. 2007. URL <http://www.sciencedirect.com/science/article/pii/S0377221706002050>.
- J. LIESIÖ, P. MILD et A. SALO : Robust portfolio modeling with incomplete cost information and project interdependencies. *European Journal of Operational Research*, 190(3):679–695, nov. 2008. ISSN 0377-2217. URL <http://www.sciencedirect.com/science/article/pii/S0377221707006558>.
- M. LUCE : Choosing to avoid: Coping with negative emotional-laden consumer decisions. *Journal of Consumer Research*, 24(4):409–433, 1998.
- T. MARCHANT et M. PIRLOT : Modern decisive wives don't wear corsets. *Journal of Multi-Criteria Decision Analysis*, 8(4):237–238, 1999. ISSN 1099-1360.
- J. MARICHAL : On sugeno integrals as an aggregation function. *Fuzzy Sets and Systems*, 114:347–365, 2000.
- M. MARTIN et W. FUERST : Effective design and use of computer decision models. *Management Information Systems Quarterly*, 8(1), déc. 1984. URL <http://aisel.aisnet.org/misq/vol8/iss1/2>.
- V. MASSOL et T. HUSTED : *JUnit in Action*. Manning Publications Co., Greenwich, CT, USA, 2003. ISBN 1930110995.

- N. MATSATSINIS et A. P. SAMARAS : MCDA and preference disaggregation in group decision support. *European Journal of Operational Research*, 130 (2):414–429, avr. 2001.
- B. MAYAG, O. CAILLOUX et V. MOUSSEAU : MCDA tools and risk analysis: the decision deck project. In C. BÉRENGUER, A. GRALL et C. GUEDES SOARES, édés : *Advances in Safety, Reliability and Risk Management*, p. 2324–2330, Troyes, France, sept. 2011. Taylor and Francis Group, London. URL <http://www.esrel2011.com/>. Sept. 18-22, 2011, Troyes, France.
- P. MEYER, J.-L. MARICHAL et R. BISDORFF : Disaggregation of bipolar-valued outranking relations. In H. A. LE THI, P. BOUVRY et T. PHAM DINH, édés : *Proc. of MCO'08 conference*, p. 204–213, Metz, France, sept. 2008. Springer. URL <http://public.enst-bretagne.fr/~pmeyer/articles/pdf/meyerMarichalBisdorff2008.pdf>.
- R. P. MINCH et G. L. SANDERS : Computerized information systems supporting multicriteria decision making. *Decision Sciences*, 17(3):395–413, 1986. ISSN 1540-5915. URL <http://onlinelibrary.wiley.com/doi/10.1111/j.1540-5915.1986.tb00233.x/abstract>.
- V. MOUSSEAU, L. DIAS et J. FIGUEIRA : Dealing with inconsistent judgments in multiple criteria sorting models. *4OR*, 4(3):145–158, 2006.
- V. MOUSSEAU, L. DIAS, J. FIGUEIRA, C. GOMES et J. CLÍMACO : Resolving inconsistencies among constraints on the parameters of an MCDA model. *European Journal of Operational Research*, 147(1):72–93, 2003.
- V. MOUSSEAU, J. FIGUEIRA et J. NAUX : Using assignment examples to infer weights for ELECTRE TRI method: Some experimental results. *European Journal of Operational Research*, 130(2):263–275, avr. 2001. URL <http://www.sciencedirect.com/science/article/pii/S0377221700000412>.
- V. MOUSSEAU et R. SŁOWIŃSKI : Inferring an ELECTRE TRI model from assignment examples. *Journal of Global Optimization*, 12(2):157–174, 1998.
- V. MOUSSEAU, R. SŁOWIŃSKI et P. ZIELNIEWICZ : A user-oriented implementation of the ELECTRE TRI method integrating preference elicitation support. *Computers & Operations Research*, 27(7-8):757–777, 2000.
- E. NATIVIDADE-JESUS, J. COUTINHO-RODRIGUES et C. H. ANTUNES : A multicriteria decision support system for housing evaluation. *Decision Support Systems*, 43(3):779–790, avr. 2007. ISSN 0167-9236. URL <http://www.sciencedirect.com/science/article/pii/S0167923606002156>.

Bibliographie

- A. NGO THE et V. MOUSSEAU : Using assignment examples to infer category limits for the ELECTRE TRI method. *JMCDA*, 11(1):29–43, nov. 2002. URL <http://www3.interscience.wiley.com/cgi-bin/abstract/99016955/START>.
- B. NORTON, R. COSTANZA et R. C. BISHOP : The evolution of preferences: Why ‘sovereign’ preferences may not lead to sustainable policies and what to do about it. *Ecological Economics*, 24(2–3):193–211, fév. 1998. ISSN 0921-8009. URL <http://www.sciencedirect.com/science/article/pii/S0921800997001432>.
- W. OUERDANE : *Multiple Criteria Decision Aiding : a Dialectical Perspective*. Thèse de doctorat, University of Paris Dauphine, France, 2009.
- W. OUERDANE, N. MAUDET et A. TSOUKIÀS : Argumentation theory and decision aiding. In M. EHRGOTT, J. R. FIGUEIRA et S. GRECO, édés : *Trends in Multiple Criteria Decision Analysis*, vol. 142 de *International Series in Operations Research & Management Science*, p. 177–208. Springer US, 2010. ISBN 978-1-4419-5904-1. URL <http://www.springerlink.com/content/v113670t17526v1q/abstract/>.
- J. PAYNE, J. BETTMAN et D. SCHKADE : Measuring constructed preferences: Towards a building code. *Journal of Risk and Uncertainty*, 19(1-3):243–270, déc. 1999.
- P. PERNY : *Modélisation, agrégation et exploitation de préférences floues dans une problématique de rangement*. Thèse de doctorat, Université Paris-Dauphine, 1992.
- L. D. PHILLIPS et C. A. BANA E COSTA : Transparent prioritisation, budgeting and resource allocation with multi-criteria decision analysis and decision conferencing. *Annals of Operations Research*, 154(1):51–68, oct. 2007. ISSN 0254-5330. URL <http://www.springerlink.com/content/8060860880u602t7/?p=18c91ea3af924ae8b73045057c9c5d02&pi=3>.
- K. R. POPPER : *The logic of scientific discovery*. Hutchinson, London, 1959.
- I. RAHWAN et G. R. SIMARI, édés. *Argumentation in Artificial Intelligence*. Springer, 2009. ISBN 978-0-387-98196-3. URL <http://www.springer.com/computer/ai/book/978-0-387-98196-3>.
- V. R. RAO, V. MAHAJAN et N. P. VARAIYA : A balance model for evaluating firms for acquisition. *Management Science*, 37(3):331–349, 1991. URL <http://mansci.journal.informs.org/cgi/content/abstract/37/3/331>.

- B. ROY : Classement et choix en présence de points de vue multiples: La méthode ELECTRE. *Revue d'Informatique et de Recherche Opérationnelle*, 2(8 (série verte n° 1)):57–75, 1968. ISSN 1290-3868.
- B. ROY : *Méthodologie multicritère d'aide à la décision*. Economica, Paris, 1985.
- B. ROY : Main sources of inaccurate determination, uncertainty and imprecision. *Mathematical and Computer Modelling*, 12(10/11):1245–1254, 1989.
- B. ROY : Science de la décision ou science de l'aide à la décision? Rap. tech., Cahier du LAMSADE N° 97, Université Paris-Dauphine, Paris, 1990. URL <http://www.lamsade.dauphine.fr/spip.php?rubrique45>.
- B. ROY : Science de la décision ou science de l'aide à la décision? *Revue Internationale de Systémique*, 6:497–529, 1992.
- B. ROY : Decision science or decision-aid science? *European Journal of Operational Research*, 66:184–203, 1993.
- B. ROY : Paradigms and challenges. In J. FIGUEIRA, S. GRECO et M. EHRGOTT, édés : *Multiple Criteria Decision Analysis: State of the Art Surveys*, p. 3–26. Springer Verlag, Boston, Dordrecht, London, 2005a. URL <http://www.springeronline.com/sgw/cda/frontpage/0,11855,5-165-22-34954528-0,00.html>.
- B. ROY, M. ALOULOU et R. KALAI, édés. *Robustness in OR-DA*. Num. 7 de Annales du Lamsade. Lamsade, 2007. URL http://www.lamsade.dauphine.fr/mcda/biblio/PDF/LAMSADE_AN7.pdf.
- B. ROY et P. BERTIER : La méthode ELECTRE II - une application au média-planning. In R. M, éd. : *OR'72*, p. 291–302. North-Holland Publishing Company, 1973.
- B. ROY et D. BOUYSSOU : *Aide Multicritère à la Décision : Méthodes et Cas*. Economica, Paris, 1993. URL <http://www.lamsade.dauphine.fr/~bouyssou/pub.html>.
- B. ROY et P. VINCKE : The case of the vanishing optimum revisited again. *Journal of Multi-Criteria Decision Analysis*, 7:351, 1998.
- B. ROY : A propos de robustesse en recherche opérationnelle et aide à la décision. In J.-C. BILLAUT, A. MOUKRIM et E. SANLAVILLE, édés : *Flexibilité et robustesse en ordonnancement*, p. 35–50. Lavoisier, 2005b. URL <http://www.lavoisier.fr/notice/fr2746210280.html>.

Bibliographie

- A. SALO : Interactive decision aiding for group decision support. *European Journal of Operational Research*, 84(1):134–149, 1995.
- A. SALO, J. KEISLER et A. MORTON : *Portfolio Decision Analysis*. Springer-Verlag New York Inc., juil. 2011. ISBN 1441999426.
- A. SCHÄRLIG : The case of the vanishing optimum. *Journal of Multi-Criteria Decision Analysis*, 5:160–164, 1996.
- Y. SISKOS, E. GRIGOROUDIS et N. MATSATSINIS : UTA methods. In J. FIGUEIRA, S. GRECO et M. EHRGOTT, édés : *Multiple Criteria Decision Analysis: State of the Art Surveys*, p. 297–344. Springer Verlag, Boston, Dordrecht, London, 2005. URL <http://www.springeronline.com/sgw/cda/frontpage/0,11855,5-165-22-34954528-0,00.html>.
- P. SLOVIC et A. TVERSKY : Who accepts savage’s axiom? *Behavioral Science*, 19:368–373, 1974.
- P. SLOVIC : The construction of preference. *American Psychologist*, 50(5): 364–371, 1995. ISSN 1935-990X.
- T. SPENGLER, J. GELDERMANN, S. HÄHRE, A. SIEVERDINGBECK et O. RENTZ : Development of a multiple criteria based decision support system for environmental assessment of recycling measures in the iron and steel making industry. *Journal of Cleaner Production*, 6(1):37–52, 1998. ISSN 0959-6526. URL <http://www.sciencedirect.com/science/article/pii/S0959652697000486>.
- K. E. STANOVICH et R. F. WEST : Discrepancies between normative and descriptive models of decision making and the Understanding/Acceptance principle. *Cognitive Psychology*, 38(3):349 – 385, mai 1999. ISSN 0010-0285. URL <http://www.sciencedirect.com/science/article/pii/S0010028598907009>.
- C. R. SUNSTEIN : Preferences and politics. *Philosophy & Public Affairs*, 20(1):3–34, 1991. ISSN 0048-3915. URL <http://www.jstor.org/stable/2265216>.
- R. SŁOWIŃSKI, S. GRECO et B. MATARAZZO : Axiomatization of utility, outranking and decision-rule preference models for multiple-criteria classification problems under partial inconsistency with the dominance principle. *Control and Cybernetics*, 31(4), 2002.

- J. TEGHEM, C. DELHAYE et P. L. KUNSCH : An interactive decision support system (IDSS) for multicriteria decision aid. *Mathematical and Computer Modelling*, 12(10-11):1311–1320, 1989. ISSN 0895-7177. URL <http://www.sciencedirect.com/science/article/pii/0895717789903701>.
- T. TERVONEN : JSMAA: open source software for SMAA computations. *International Journal of Systems Science*, p. 1–13, fév. 2012. ISSN 0020-7721. URL <http://www.tandfonline.com/doi/abs/10.1080/00207721.2012.659706>.
- T. TERVONEN, J. R. FIGUEIRA, R. LAHDELMA, J. A. DIAS et P. SALMINEN : A stochastic method for robustness analysis in sorting problems. *European Journal of Operational Research*, 192(1):236 – 242, 2009. ISSN 0377-2217. URL <http://www.sciencedirect.com/science/article/pii/S0377221707009277>.
- A. TVERSKY, S. SATTATH et P. SLOVIC : Contingent weighting in judgment and choice. *Psychological Review*, 95(3):371–384, 1988. ISSN 0033-295X. URL <http://psycnet.apa.org/journals/rev/95/3/371/>.
- A. TVERSKY et E. SHAFIR : Choice under conflict: The dynamics of deferred decision. *Psychological Science*, 3(6):358–361, nov. 1992. ISSN 0956-7976, 1467-9280. URL <http://pss.sagepub.com/content/3/6/358>.
- G. van VALKENHOEF, T. TERVONEN, T. ZWINKELS, B. de BROCK et H. HILLEGE : ADDIS: a decision support system for evidence-based medicine. *Decision Support Systems*, 2012. To appear.
- P. VINCKE : Robust solutions and methods in decision aid. *Journal of Multi-Criteria Decision Analysis*, 8(3):181–187, 1999.
- P. VINCKE : *L'aide multicritère à la décision*. Ellipses Marketing, mai 1998. ISBN 2729889426.
- VOLTAIRE : *La Princesse de Babylone*. J'ai lu, texte intégral éd'n, mars 2000. ISBN 2290304964.
- J. WALLENIS, J. S. DYER, P. C. FISHBURN, R. E. STEUER, S. ZIONTS et K. DEB : Multiple criteria decision making, multiattribute utility theory: Recent accomplishments and what lies ahead. *Management Science*, 54(7):1336 –1349, juil. 2008. URL <http://mansci.journal.informs.org/content/54/7/1336.abstract>.

Bibliographie

- J. ZHENG, O. CAILLOUX et V. MOUSSEAU : Constrained multicriteria sorting method applied to portfolio selection. In R. I. BRAFMAN, F. S. ROBERTS et A. TSOUKIÀS, édés : *Algorithmic Decision Theory*, vol. 6992 de *Lecture Notes in Computer Science*, p. 331–343, Rutgers, United States, oct. 2011. Springer Berlin / Heidelberg. ISBN 978-3-642-24872-6. URL <http://adt2011.org/>.
- S. ZIONTS : The case of the vanishing optimum revisited. *Journal of Multi-Criteria Decision Analysis*, 6:247, 1997.
- C. ZOPOUNIDIS et M. DOUMPOS : A multicriteria decision aid methodology for the assessment of country risk. *European Research on Management and Business Economics*, 3(3):13–33, 1997.
- C. ZOPOUNIDIS et M. DOUMPOS : A preference disaggregation decision support system for financial classification problems. *European Journal of Operational Research*, 130(2):402–413, avr. 2001.
- C. ZOPOUNIDIS et M. DOUMPOS : PREFDIS: a multicriteria decision support system for sorting decision problems. *Computers & Operations Research*, 27(7-8):779–797, juin 2000. ISSN 0305-0548. URL <http://www.sciencedirect.com/science/article/pii/S0305054899001185>.

Annexes

A Calcul du surclassement dans d'autres modèles ÉLECTRE

A.1 ÉLECTRE IV

Le modèle ÉLECTRE IV est une variante du modèle ÉLECTRE I (Roy, 1968), qui utilise une autre manière de calculer la relation de discordance, mais que nous ne détaillons pas (Figueira *et al.*, 2005b). La lettre v dans le nom du modèle fait référence aux seuils de véto. Le livre de Roy et Bouyssou (1993) présente le modèle ÉLECTRE IV sous le nom ÉLECTRE I. Il ne doit pas être confondu avec le modèle ÉLECTRE IV, que nous ne présentons pas.

ÉLECTRE IV n'utilise pas de seuils p_j, q_j . Ses paramètres sont les poids w_j , les seuils de véto v_j , et un seuil de majorité λ . Il s'appuie sur une relation de concordance $\Psi^{\text{coalitions}}$ et sur une relation de discordance D^{binaire} définies comme suit.

$$\Psi^{\text{coalitions}}(a, b) = \sum_{\{j | g_j(a) \geq g_j(b)\}} w_j.$$

Cette relation est associée au nom « coalitions » car elle compte la somme des poids d'une coalition de critères en faveur d'un surclassement. La relation $\Psi^{\text{coalitions}}$ correspond à la relation Ψ lorsque K_j est calculée avec des seuils $p_j = q_j = 0$.

$$D^{\text{binaire}}(a, b) = \begin{cases} 1 & \Leftrightarrow \exists j \in \mathcal{J} \mid g_j(b) - g_j(a) > v_j, \\ 0 & \text{sinon.} \end{cases}$$

$$S^{\text{agrégation}, \lambda}(a, b) = \begin{cases} 1 & \Leftrightarrow \Psi^{\text{coalitions}}(a, b) \geq \lambda \text{ et } D^{\text{binaire}}(a, b) = 0, \\ 0 & \text{sinon.} \end{cases}$$

Le calcul de la relation de surclassement du modèle MR Sort est très proche du calcul proposé par ÉLECTRE IV. Ce dernier n'avait toutefois pas pour objectif de procéder à un tri et n'utilisait pas de profils. Le modèle MR Sort peut donc être vu comme un mélange de ÉLECTRE TRI et de ÉLECTRE IV.

A.2 ÉLECTRE II

ÉLECTRE II (Roy et Bertier, 1973) construit deux relations de surclassement binaires, S^- et S^* . La relation S^- indique un surclassement éventuellement incertain, tandis que la relation S^* n'indique que les surclassements associés à une forte crédibilité. Ces deux relations sont emboîtées : si aS^*b , alors aS^-b . Le modèle utilise un seuil de majorité faible λ^- et un seuil de majorité fort λ^* , $\lambda^- < \lambda^*$, et un ensemble de seuils de véto faibles v_j^- et forts v_j^* , $v_j^* \leq v_j^-$.

$$S^-(a, b) = \begin{cases} 1 & \Leftrightarrow \begin{cases} \Psi^{\text{coalitions}}(a, b) \geq \lambda^- \\ \text{et} \\ D^{\text{binaire}}(a, b) = 0 \\ \text{et} \\ \Psi^{\text{coalitions}}(a, b) \geq \Psi^{\text{coalitions}}(b, a), \end{cases} \\ 0 & \text{sinon,} \end{cases}$$

où D^{binaire} utilise les valeurs de véto v_j^- . La relation $S^*(a, b)$ est définie de manière similaire mais utilise les valeurs λ^* et v_j^* .

Ces relations ressemblent à la relation $S^{\text{agrégation}, \lambda}$ mais contiennent une condition supplémentaire. La condition $\Psi^{\text{coalitions}}(a, b) \geq \Psi^{\text{coalitions}}(b, a)$ a été ajoutée de manière à éviter autant que possible une situation symétrique où aSb et bSa .

A.3 ÉLECTRE III

ÉLECTRE III utilise les relations de concordance, discordance et surclassement flous présentées au début de cette section : $K_j, \Psi, D_j, S^{\text{simplifié}}$. Le modèle requiert les paramètres $p_j, q_j, v_j, \lambda, w_j$.

B Brève description des modèles PROMÉTHÉE I et PROMÉTHÉE II

Cette section présente succinctement les concepts principaux utilisés par la famille de modèles PROMÉTHÉE (Brans, 1982 ; Brans et Vincke, 1985). PROMÉTHÉE I propose un rangement partiel des alternatives, PROMÉTHÉE II propose un rangement complet. D'autres extensions ont été proposées (Brans et Mareschal, 2005), mais nous n'en parlerons pas ici. L'explication se compose de deux parties. Nous expliquons d'abord comment ces modèles calculent la préférence, puis comment ils l'exploitent pour obtenir un rangement.

B.1 Calcul de la préférence

PROMÉTHÉE offre cinq types de « fonctions de préférence », fonctions utilisées pour déterminer la préférence du décideur entre deux alternatives sur un critère donné. Les fonctions de préférence sur les différents critères du problème ne doivent pas nécessairement être toutes de même type. Les paramètres utilisés diffèrent selon les types. Nous présentons quatre des six types proposés classiquement. Les paramètres sont p_j , seuil de préférence et q_j , seuil d'indifférence, $p_j \geq q_j \geq 0$, et les poids $w_j \geq 0$ que nous supposons normalisés.

$$P_j^1(a, b) = \begin{cases} 1 & \Leftrightarrow g_j(a) - g_j(b) > 0, \\ 0 & \Leftrightarrow g_j(a) - g_j(b) \leq 0. \end{cases}$$

$$P_j^2(a, b) = \begin{cases} 1 & \Leftrightarrow g_j(a) - g_j(b) > q_j, \\ 0 & \Leftrightarrow g_j(a) - g_j(b) \leq q_j. \end{cases}$$

$$P_j^3(a, b) = \begin{cases} 1 & \Leftrightarrow p_j < g_j(a) - g_j(b), \\ \frac{g_j(a) - g_j(b)}{p_j} & \Leftrightarrow 0 < g_j(a) - g_j(b) \leq p_j, \\ 0 & \Leftrightarrow g_j(a) - g_j(b) \leq 0. \end{cases}$$

$$P_j^5(a, b) = \begin{cases} 1 & \Leftrightarrow p_j < g_j(a) - g_j(b), \\ \frac{(g_j(a) - g_j(b)) - q_j}{p_j - q_j} & \Leftrightarrow q_j < g_j(a) - g_j(b) \leq p_j, \\ 0 & \Leftrightarrow g_j(a) - g_j(b) \leq q_j. \end{cases}$$

La fonction de préférence globale $P : \mathbb{A} \times \mathbb{A} \mapsto [0, 1]$ indique, étant donné une paire d'alternatives (a, b) , dans quelle mesure l'alternative a est préférée à b .

$$P(a, b) = \sum_{j \in \mathcal{J}} w_j P_j(a, b),$$

où $P_j = P_j^1, P_j^2, P_j^3$ ou P_j^5 selon le type de fonction de préférence choisi.

Notons qu'à valeurs de paramètres égales, $P(a, b) = 1 - \Psi(b, a)$. C'est donc surtout dans la partie de l'exploitation que les modèles PROMÉTHÉE se distinguent des modèles ÉLECTRE.

B.2 Exploitation

Chaque alternative est ensuite associée à un flux positif (Q^+) et à un flux négatif (Q^-), qui définissent deux rangements sur l'ensemble d'alternatives.

$$Q^+(a) = \frac{1}{|A| - 1} \sum_{b \in \mathbb{A} \setminus \{a\}} P(a, b),$$

$$Q^-(a) = \frac{1}{|A| - 1} \sum_{b \in \mathbb{A} \setminus \{a\}} P(b, a).$$

Le rangement partiel final est l'intersection des deux rangements induits par les flux positifs et négatifs.

PROMÉTHÉE II introduit le calcul du flux net, les autres calculs restant inchangés. Le flux net détermine un rangement complet sur les alternatives.

$$Q(a) = Q^+(a) - Q^-(a).$$

C Remarque concernant les vétos sur critères de poids nul dans les modèles de type ÉLECTRE TRI

Le modèle ÉLECTRE TRI est habituellement décrit sous l'hypothèse implicite qu'aucun poids de critère n'est nul : sans utilisation de vétos, un tel critère serait simplement inutile et ne devrait pas être considéré dans l'ensemble \mathcal{J} . Pourtant, le raisonnement ne tient pas nécessairement dans un modèle autorisant des vétos. Deux principes d'ÉLECTRE se contredisent dans le cas où un veto est associé à un critère de poids nul. Un des principes est qu'une concordance unanime, donc une situation où $\Psi(a, b) = 1$, devrait toujours conduire à un surclassement. L'autre principe est qu'une situation où le veto s'exerce pleinement du point de vue d'au moins un critère, donc où $\exists j \in J \mid D_j(a, b) = 1$, devrait entraîner un rejet du surclassement.

Dans le modèle NCSM avec veto, un veto peut être associé à un critère qui n'est jamais nécessaire à une coalition. Dans un tel cas, le veto prendra effet et interdira l'accès de l'alternative à la catégorie à laquelle le veto est associé. Soit F un ensemble de coalitions, donc $F \subseteq 2^{\mathcal{J}}$. Appelons les coalitions dans F coalitions gagnantes. Nous disons qu'un critère $j \in \mathcal{J}$ est nécessaire à au moins une coalition, ou simplement que j est opérant, ssi il fait partie d'au moins une coalition $J \in F$ telle que si on retire le critère, la coalition n'est plus gagnante. Formellement, le critère est dit opérant ssi :

$$\exists J \in F \mid j \in J \wedge J \setminus \{j\} \notin F. \quad (\text{critère opérant})$$

Nous souhaitons autoriser l'utilisation de poids nuls dans le modèle MR Sort. Cela nous sera utile lors de l'application du modèle à un contexte impliquant plusieurs décideurs : sans pour autant abandonner l'hypothèse commode que tous les décideurs partagent le même ensemble de critères, il nous est possible de représenter une situation où certains décideurs n'attribuent aucune importance à certains critères. Cela rend également le modèle MR Sort plus proche conceptuellement de NCSM, ce dernier autorisant des critères non nécessaires à une quelconque coalition. Enfin, interdire les poids nul sans interdire les poids très faibles serait étonnant : dans beaucoup de situations,

C Remarque concernant les véto sur critères de poids nul

il est possible d'assurer qu'un critère ne soit jamais nécessaire à la moindre coalition en lui donnant un poids suffisamment faible plutôt qu'un poids nul. Considérons par exemple les coalitions déterminées par les valeurs suivantes : $w_1 = 0,4; w_2 = 0,4; w_3 = 0,2; \lambda = 0,8$. Dans un tel cas, le troisième critère ne permet jamais d'obtenir une coalition gagnante.

Dès lors que nous autorisons un critère de poids nul, il faut décider de la valeur du surclassement S lorsqu'un véto est associé à un tel critère. Par conformité avec le modèle NCSM, nous avons choisi de faire jouer le véto normalement dans MR Sort, même associé à un critère de poids nul (voir (affectation MR Sort) page 50).

D Données publiées en format électronique

Nous avons souhaité publier les données correspondant aux expériences effectuées, autant que possible, en format électronique. Disposer de colonnes de chiffres imprimées sur papier n'est pas très utile, alors que les fichiers électroniques peuvent être réutilisés et analysés plus simplement. Les données sont publiées au format standard XMCD (version 2.2.0).

Voici la liste des données publiées.

1. Les données de l'exemple illustrant la méthode de convergence de groupe décrite à la section 3.1. L'exemple est décrit à la section 3.1.6. Le premier fichier contient le contexte du problème et donne les performances des alternatives, le deuxième contient les exemples d'affectation.
2. Les données de l'application de construction d'une échelle de risque pour des produits polluants, décrite à la section 3.3. Les données contiennent : le tableau de performance des vingt substances exemples, sur les huit critères considérés ; les neuf fois vingt exemples d'affectation donnés par chaque participant.
3. Les données correspondant à l'exemple décrit à la section 4.4.
 - 1/ Les performances des dix alternatives utilisées dans les exemples.
 - 2/ Les crédibilités initiales associées aux dix alternatives exemples.
 - 3/ Les performances des cent étudiants à classer.
 - 4/ Les affectations des cent étudiants, lors de la première itération, aux niveaux de crédibilité 7 et 6.
 - 5/ Les affectations des cent étudiants, lors de la deuxième itération, aux niveaux de crédibilité 7, 6 et 5.
 - 6/ Les affectations des cent étudiants, lors de la troisième itération, aux niveaux de crédibilité 7, 6 et 5.
4. Les données correspondant à l'exemple de choix de portefeuille décrit à la section 5.4.
 - 1/ Les trente exemples de projets.

D Données publiées en format électronique

- 2/ Les performances des cent projets à utiliser pour constituer le portefeuille.
- 3/ Le premier modèle trouvé et les affectations des cent projets correspondant au choix de ce modèle.
- 4/ Le deuxième modèle trouvé et les affectations des cent projets correspondant au choix de ce modèle.
- 5/ Le troisième modèle trouvé et les affectations des cent projets correspondant au choix de ce modèle.
5. Le flux de données présenté à la section 6.5. Les instructions d'utilisation sont disponibles à l'adresse <http://www.decision-deck.org/diviz/workflow.ressArticle.html>.
6. Le modèle de données et les exemples présentés au chapitre 7.