HAL
open science

Systèmes interactifs d'aide à l'élaboration de plannings de travail de personnel : Contraintes, aide à la décision, représentation combinatoire des préférences, équité et résolution par décomposition arborescente et par consistance

Rémy-Robert Joseph

To cite this version:

Rémy-Robert Joseph. Systèmes interactifs d'aide à l'élaboration de plannings de travail de personnel : Contraintes, aide à la décision, représentation combinatoire des préférences, équité et résolution par décomposition arborescente et par consistance. domain_stic.conf. Université Joseph-Fourier - Grenoble I, 2003. Français. NNT : . tel-00004651v1

HAL Id: tel-00004651

https://theses.hal.science/tel-00004651v1

Submitted on 8 Mar 2004 (v1), last revised 25 Mar 2004 (v2)

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

UNIVERSITÉ JOSEPH FOURIER-GRENOBLE I
SCIENCES & GÉOGRAPHIE

THÈSE

pour obtenir le grade de

DOCTEUR DE L'UNIVERSITÉ JOSEPH FOURIER

Discipline : Informatique : Systèmes d'Information de Santé, Cognition

Formation Doctorale : Méthodes et Instrumentation pour la Médecine et la Biologie

Présentée et soutenue publiquement

par

Rémy-Robert, Alexandre JOSEPH

Le 07 novembre 2003

Systèmes Interactifs d'Aide à l'Élaboration de Plannings de Travail de Personnel

Contraintes, Aide à la Décision, Représentation Combinatoire des Préférences, Équité et Résolution par Décomposition Arborescente et par Consistance.

Composition du Jury :

Présidente :	Mme Catherine GARBAY
Directeur :	M. Georges WEIL
Co-Directeur :	M. Gerd FINKE
Rapporteurs :	M. Denis BOUYSSOU
	M. Gérard VERFAILLIE

Thèse préparée au sein du Laboratoire **TIMC – Institut IMAG**

À mes parents et mes frères, avec affection.

À Claude, avec tendresse.

Remerciements

En débutant cette thèse, je ne pensais pas découvrir une telle diversité dans les travaux se rapprochant de mon sujet. J'ai ainsi pu m'imprégner d'éthique philosophique et de choix social ; Un monde qui m'était aussi inconnu qu'attractif. Cet univers complète à bien des égards ma formation première basée sur l'aspect purement technique des problèmes de décision (informatique, mathématiques discrètes et optimisation).
Par conséquent, je remercie Georges WEIL, praticien hospitalier et maître de conférence à l'Université Joseph Fourier, président-directeur général de la société Equitime SA et directeur de cette thèse, de m'avoir laissé autant de liberté dans mon travail ; Il a su me sensibiliser sur la nécessité de mieux intégrer l'humain dans les problèmes de planification.

Je tiens à remercier particulièrement Denis BOUYSSOU, directeur de recherche au LAMSADE, et Gérard VERFAILLIE, directeur de recherche au LAAS, pour avoir bien voulu être rapporteurs de cette thèse malgré le peu de temps qui leur était alloué. Leurs travaux ont beaucoup influencé les orientations que j'ai prises dans ce travail. Merci pour l'intérêt qu'ils m'ont manifesté en retour.

Ma gratitude va également à Gerd FINKE, professeur à l'Université Joseph Fourier, responsable de l'équipe Recherche Opérationnelle du laboratoire LEIBNIZ et co-directeur de cette thèse, pour l'attention qu'il a apportée à mon travail durant ces cinq dernières années.

Catherine GARBAY, chargée de recherche et directrice adjointe du laboratoire TIMC, m'a fait l'honneur d'accepter de présider le jury de cette thèse. Je la remercie sincèrement de son accueil dans son équipe Systèmes Intégrés Cognitifs ; Cela m'a permis de réaliser cette thèse dans d'excellentes conditions.

Un clin d'œil amical au professeur Hervé RAYNAUD, lequel m'a laissé entrevoir, entre autre, les richesses que peuvent renfermer le choix social et la psychologie pour la compréhension du monde qui nous entoure.

Je tiens à témoigner ma reconnaissance à tous ceux qui ont participé de près ou de loin à la réalisation de ce travail, en particulier mes premiers lecteurs Claude ÉLISÉE et Peter CHAN qui ont fait un travail de relecture très perspicace. Merci beaucoup pour la patience dont ils ont fait preuve. Grâce à eux ce manuscrit est devenu plus agréable à lire.

J'adresse toute ma sympathie à la jeune équipe dynamique de la société Equitime. Nos échanges m'ont permis de mieux apprécier les difficultés auxquelles sont confrontées les entreprises lors de la planification. Je remercie également Kamel HEUS sans qui cette aventure n'aurait pas eu lieu.

Cette thèse ne serait pas complète sans les encouragements durant toutes ces années de mes amis du DEA ROCO 1998 : Amel, Karine, Nedjima, Olfa, Sophiann, Yasmina, Ameur, Denis, Emilio, Haris et Hicham.

Ces années seraient sans doute moins agréables sans tous les (ex-) doctorants du laboratoire TIMC : Anne, Anne-Sophie, Catherine, Christine, Sonia, Antoine, Christophe, Éric, Mathieu, Olivier, Vincent … ainsi que les membres du département de Mathématiques Discrètes du laboratoire LEIBNIZ. L'environnement de travail est essentiel à l'épanouissement

d'un jeune chercheur. Je souhaite remercier ici toutes les personnes qui font du laboratoire TIMC un cadre agréable et stimulant.

J'adresse des remerciements tout particuliers à mes amis de longue date : Karol (Hervé), Angye, Ti-Kako (Fabrice), Clifford, Félix et Jean-Claude qui n'ont pas hésité à me motiver et m'encourager jusqu'au bout. Qu'ils reçoivent ici le témoignage de ma sincère amitié.

Table des Matières

Remerciements 5
Table des Matières 7
Table des Figures 11
Abréviations et Indications de Lecture 13
Introduction 15
Modélisation et Processus de Décision 19
1. Problèmes d'Élaboration de Plannings de Travail de Personnel (PEPTP) 21
1.1. Généralités 21
1.1.1. Qu'est ce qu'un Planning ? 21
1.1.2. Unités et Acteurs 21
1.1.3. A Quoi Sert un Planning ? Les Enjeux 22
1.2. Aspects Combinatoire et Négociation des Plannings 23
1.2.1. Aspect Combinatoire 23
1.2.2. Quelques Organisations 23
1.2.3. Comment Sont Évalués les Plannings ? : Aspect Négociation 24
1.2.4. Qui Peut se Charger de l'Élaboration ? 25
1.3. Problématique Décisionnelle 25
2. Modèles Théoriques de Base et État de l'Art des Modèles en Élaboration de Plannings 27
2.1. Modélisation Générale : Quelques Rappels 27
2.1.1. Diversité et Complexité des Problèmes 27
2.1.2. Graphes et Problèmes 32
2.1.3. Les Systèmes d'Inégalités en Nombres Mixtes 33
2.1.4. Les Problèmes sur les Systèmes de Contraintes 36
2.2. Panorama des Problèmes d'Élaboration de Plannings de Travail de Personnel 44
2.2.1. Concepts et Notation 45
2.2.2. Typologie des Problèmes d'Élaboration de Plannings de Travail 50
2.2.3. Problèmes d'Élaboration de Planning de Type Affectation (PEPTA) 53
2.2.4. Problèmes d'Élaboration de Plannings de Type Affectation Généralisée (PEPTAG) 59
2.2.5. Problèmes d'Élaboration de Plannings de Type Recouvrement d'Ensemble Généralisé (PEPTREG) 64
2.2.6. Problèmes d'Élaboration de Planning de Type Implicite (PEPTI) 66
2.2.7. Génération d'Horaires de Travail 67
2.2.8. Problèmes de Dimensionnement d'Effectif 69
2.3. Bilan Préférentiel et Perspectives 70
2.3.1. Modélisations Usuelles des Préférences et Synthèse Préférentielle 70
2.3.2. Préférences et Aide à la Décision : l'Interactivité. 71
2.3.3. Complémentarité entre Modélisation des Préférences et Interactivité ? 72
2.3.4. Représentation Implicite des Préférences 72
2.3.5. Modélisations de l'Équité. 72
3. Avancées en Modélisation : Aide à la Décision et Préférences Individuelles 75
3.1. Préliminaires : Aide Multicritère à la Décision et Choix Social 75
3.1.1. Qu'est-ce que l'Aide Multicritère à la Décision ? 76
3.1.2. Qu'est-ce que la Théorie du Choix Social ? 76

3.1.3. Intelligence Artificielle et Traitement de l'Information 77
3.1.4. Une Présentation Commune Générique 77
3.2. Modèles, Processus de Décision et Aide à la Décision 78
3.2.1. Conception de Modèles et Aide à la Décision 78
3.2.2. Systèmes Interactifs d'Aide à la Décision et Processus de Décision 81
3.2.3. Problématiques d'Aide à la Décision 85
3.2.4. Conclusion 91
3.3. Modèles de Base des Préférences 91
3.3.1. Les Relations Binaires de Préférences 92
3.3.2. Préférences Binaires et Structures de Préférence 95
3.3.3. Fonction-Critères, Graduations 101
3.3.4. Les Relations Non-Binaires de Préférence 104
3.4. Aider à Choisir avec une Relation Binaire de Préférence 105
3.4.1. Optimalité et Maximalité 106
3.4.2. Redondance et Approximation Exprimées par la Stabilité Maximale 107
3.4.3. Au-Delà de l'Optimalité et de la Maximalité 108
3.4.4. Choisir avec Prudence : Optimalité Relative et Maximalité Relative 111
3.4.5. Redondance Exprimée par l'Indiscernabilité 117
3.4.6. Conclusion et Compléments 119
4. Avancées en Modélisation : Profils, Information Interindividuelle et Agrégation 123
4.1. Profils et Informations Interindividuelles 123
4.1.1. Le Profil Simple 123
4.1.2. Informations Préférentielles Interindividuelles 126
4.1.3. Profils Basés sur les Comparaisons Interpersonnelles Étendues 128
4.2. Aider à Choisir avec un Profil : Solutions de Meilleur Compromis 133
4.2.1. Choix Collectif et Agrégation 133
4.2.2. Choisir une Règle de Décision Collective Adéquate par Analyse Axiomatique 141
5. Contraintes et Représentation Implicite des Préférences 147
5.1. Introduction 147
5.1.1. Comment Exprimer des Préférences sur des Solutions Alternatives non Explicitement Données ? 147
5.1.2. Modélisation de Préférences par l'Intermédiaire de Contraintes Souples 147
5.1.3. Les Grandes Lignes du Chapitre 148
5.2. Contraintes Souples dédiées aux Préférences Binaires à Attitude Fondamentales en Extension 148
5.2.1. Contraintes Souples Dédiées aux Préférences Binaires Simples 149
5.2.2. Contraintes Souples Dédiées aux Préférences Binaires Étendues 150
5.2.3. Les VC-Contraintes Souples Étendues 152
5.2.4. Quelques Propriétés des PB-Contraintes Souples 152
5.3. Systèmes de Contraintes Basés sur les Préférences et Problèmes d'Aide à la Décision 154
5.3.1. Contraintes Souples pour Règles d'Agrégation et Axiomes 154
5.3.2. Modèles de Préférence et Systèmes de Contraintes 156
5.3.3. Problèmes d'Aide au Choix et Systèmes de Contraintes Basés sur les Préférences 157
5.3.4. Choix et Incohérence dans les Systèmes de Contraintes Basés sur les Préférences 158
5.3.5. Choix, Consistance et Systèmes de Contraintes Basés sur les Préférences 161
6. L'Équité comme Outil d'Aide à l'Évaluation en Planification 165
6.1. Introduction 165
6.2. Dominance de Lorenz et Généralisations pour l'Évaluation de l'Équité dans les Problèmes d'Allocation de Ressources 166
6.2.1. Un Cas d'École : Le Problème de Pure Répartition 166

6.2.2. Quelques Outils Usuels de Mesure des Inégalités 169
6.2.3. La Dominance de Lorenz.....172
6.2.4. Quelques Extensions de la Dominance de Lorenz.....177
6.2.5. Raffinements : Dominance de Lorenz et Relation de Préférence Valuée.....178
6.2.6. Relaxations d'Informations de Faisabilité 179
6.2.7. Le Cas des Ressources Multidimensionnelles 180
6.2.8. Dominance de Lorenz et Autres Critères.....183
6.3. Équité, Efficacité et Information Préférentielle Étendue Ordinale.....187
6.3.1. Dominance de Lorenz et Comparaisons Interpersonnelles de Type Sympathie Étendue 187
6.3.2. Équité et Efficacité pour l'Agrégation de Relation Binaire de Préférence Étendue 189
6.3.3. Efficacité et Équité à partir d'un Profil Quelconque de Relations Binaires de Préférence Étendues.....193
6.3.4. Théorie des Jeux et Problèmes de Répartition.....197
6.3.5. Allocation de Ressources, Préférences Individuelles et Équité 197
6.4. Équité et Efficacité pour l'Évaluation des Plannings de Travail 200
6.4.1. Modèles Préférentiels des Problèmes d'Élaboration de Plannings de Personnel200
6.4.2. Agrégation Basée sur le Rôle des Acteurs Principaux.....200

Résolution Algorithmique : Optimisation Combinatoire Basée sur les Préférences …207
7. Algorithmes de Résolution de Problèmes d'Aide à la Décision Basés sur les Préférences et les Contraintes 209
7.1. Résolution de Problèmes de Choix Basés sur les Préférences et les Contraintes 209
7.1.1. Les Algorithmes de Gavanelli 209
7.1.2. Trouver une Solution Maximale par Branch and Bound 212
7.1.3. Énumérer Tous les Éléments d'un Ensemble de Choix Maximisant Stable Maximal 212
7.1.4. Au-Delà des Préordres Partiels 214
7.2. Programmation MultiObjective et Dominance de Lorenz.....215
7.2.1. Programmation par Buts et Dominance de Lorenz.....215
7.2.2. Polyèdre Convexe et Dominance de Lorenz.....219
7.3. Problèmes de Consistance et Propriétés des Contraintes Souples.....222
7.3.1. Décomposition et Filtrage Adaptable pour les PB-Contraintes Souples 222
7.3.2. Filtrage pour les VC-Contraintes Souples 224
7.4. Consistance pour les Contraintes Globales Souples de Quelques Règles d'Agrégation 225
7.4.1. Global Inverse Consistance pour la Dominance de Pareto sur un Profil de Vrai-Critères.....225
7.4.2. Inverse Consistance Adaptable pour la Règle de Pareto.....229
7.4.3. Consistance et Dominance de Lorenz.....232
7.4.4. Règle de Pareto comme Outil d'Agrégation des Préférences Individuelles sur les Positions.....242
7.4.5. Consistance et Règle d'Équité de Suppes-Sen.....243
8. Application sur un PEPTAG : Résultats et Performances 257
8.1. Variables et Contraintes du Prototype 257
8.1.1. Contrainte d'Affectation Ponctuelle 257
8.1.2. Contrainte de Charges.....257
8.1.3. Contrainte de Changement d'Étiquettes 258
8.1.4. Contrainte de Fréquence des Étiquettes.....258
8.1.5. Contrainte de Disponibilité des Agents 259
8.1.6. Contrainte de Séquences d'Étiquettes.....260
8.2. Modèle de Préférence 260
8.2.1. Contrainte de Dénombrement de Séquences d'Étiquettes 260

8.2.2. Agrégation par la Dominance de Pareto 261
8.3. Résolution 261
8.3.1. Modes de Résolution 261
8.3.2. Stratégies d'Évaluation 262
8.3.3. Stratégies de Choix et de Séparation pour la Résolution Arborescente 263
8.4. Expérimentations 264
8.4.1. Trois Exemples Traités 264
8.4.2. Influence des Choix de la Variable et de la Valeur sur la Résolution 265
8.4.3. Recherche de Solutions Maximales 266
8.4.4. Discussion 266

Conclusions et Perspectives 269

Annexes 275
1. Annexe A1 : Description de Quelques Contraintes Globales de la Littérature 276
2. Annexe A2 : Rappel sur les Niveaux de Consistance Inverse 277
2.1. Préliminaires 277
2.1.1. Quelques Définitions 277
2.2. Consistance Inverse et Problème d'Appartenance 278
2.2.1. L'Arc-Consistance 278
2.2.2. Real-Full-Lookahead et Forward-Checking 280
2.3. Inverse-Consistance et Contraintes Globales 280
2.4. Consistance Inverse en Programmation Entière 282
3. Annexe B1 : Preuves de Propositions du Chapitre 3 283
4. Annexe B2 : Preuves de Propositions du Chapitre 6 288
5. Annexe B3 : Preuves de Propositions du Chapitre 7 289

Bibliographie 293

Index des Noms 313

Index Terminologique 317

Résumé / Abstract 320

Table des Figures

Figure 1 : Schéma général de l'environnement évaluationnel des plannings de travail.....24
Figure 2 : Quelques liens de complexité entre problèmes.....30
Figure 3 : Représentation graphique des dépendances entre contraintes (récursivité).....41
Figure 4 : Exemple d'une courbe de charge d'un jour des Aéroports de Paris (ADP) sur 111 périodes, et de trois vacations réalisables (PARTOUCHE [1998]).....48
Figure 5 : Différents paramètres d'une vacation *v*. 49
Figure 6 : Classification inspirée par les activités, de quelques PEPTP (Déduite de la figure 1.3 de PARTOUCHE [1998]).....50
Figure 7 : Relations entre les types de problèmes d'élaboration de plannings de travail de personnel, induite par la classification inspirée des modèles. 53
Figure 8 : Charge exprimée par une fonction de pénalité de charge (MILLER *et al.* [1976]). ..56
Figure 9 : Représentation UML de la relation d'affectation entre les jours, les employés, les étiquettes et les compétences.....60
Figure 10 : Les SIADs au sein des systèmes de traitement de l'information (d'après LÉVINE & POMEROL [1989]).....81
Figure 11 : Équivalence entre graphe orienté simple réflexif et relation binaire de préférence sur un ensemble fini.....94
Figure 12 : Modèle du critère d'intervalle.....103
Figure 13 : Exemple de circuit de préférences strictes.....104
Figure 14 : Quelques exemples de graphes servant de support aux propriétés présentées. ...108
Figure 15 : Exemple de (*a*) dilemme entre le noyau et la maximalité, (*b*) graphe sans noyau et avec un ensemble maximal.....110
Figure 16 : Schéma récapitulant les relations entre la maximalité, l'optimalité, le noyau et la transitivité. 110
Figure 17 : Exemples de graphes illustrant les ensembles relativement optimaux et minimaux pour l'inclusion.....112
Figure 18 : Synthèse de la relation d'inclusion entre les propriétés de qualité globale. 117
Figure 19 : Synthèse des relations d'inclusion des propriétés de type redondance et approximation.....119
Figure 20 : Illustrations de la relation CA_1 de changement d'attitude d'un seul individu.....125
Figure 21 : Relations exploitées entre les règles de décision, les ensembles de choix et les profils, pour l'aide au choix.....135
Figure 22 : Exemple de la non-conservation de la P-acyclicité par la règle de Pareto. 138
Figure 23 : Illustration de l'incohérence sur un système de contraintes dont les solutions sont totalement préordonnées.....159
Figure 24 : Illustration de difficultés rencontrées lorsque deux ensembles maximaux sont comparés.....160
Figure 25 : Exemple de trois configurations de découpe d'un gâteau homogène.....167
Figure 26 : Ensemble des solutions réalisables du problème de pure répartition entre trois individus $\{e_1, e_2, e_3\}$: simplexe de dimension 2. 167
Figure 27 : Illustration du principe de transfert de Pigou-Dalton 169
Figure 28 : Quelques critères d'agrégation collective pour le problème de pure répartition. 171
Figure 29 : Illustration de courbes de Lorenz.....173
Figure 30 : Permutoèdres (*a*) d'ordre 3 : P_3 et (*b*) d'ordre 4 : P_4. 174

Figure 31 : (*a*) Dominance de Lorenz et Triangle de Kolm (*b*) Illustration d'une Courbe de Lorenz Discrète. 176
Figure 32 : Découpe de l'espace des critères ^{2} effectuée par (*a*) la dominance Pareto symétrique, et (*b*) la GL-dominance. 184
Figure 33 : Découpe de l'espace des critères ^{2} engendré par la dominance de Lorenz invariante aux changements d'échelle. 185
Figure 34 : Illustration de l'inter-solution transitivité des relations binaires de préférence étendues. 190
Figure 35 : Récapitulation des relations d'extension et d'inclusion entre les règles d'agrégation véhiculant des principes d'équité et d'efficacité. 192
Figure 36 : Illustration des axiomes d'équité de Hammond (*a*) et de transfert de Foster-Sen (*b*). 193
Figure 37 : Différents types d'agrégations exploités en TCS pour traiter les profils quelconques de relations binaires de préférence étendues. 193
Figure 38 : Exemple de conflit entre l'unanimité de la préférence stricte et l'équité de Suppes-Sen. 194
Figure 39 : Illustration de la non conservation de la transitivité par la règle d'agrégation de Pareto conditionnée par l'Identité. 195
Figure 40 : Illustration de la non conservation de l'inter-solution transitivité par la règle d'agrégation de Pareto conditionnée par l'Identité. 196
Figure 41 : Illustration de l'utilisation de préférences individuelles pour une meilleure évaluation des positions (Extrait de SEN [1992]). 199
Figure 42 : Quelques hiérarchies d'agrégation des préférences individuelles fournies par des quantités de ressources multidimensionnelles. 199
Figure 43 : Relation d'agrégation de la hiérarchie agrégative usuelle des PEPTP. 201
Figure 44 : Représentations de 3 horaires de travail qui diffèrent par leur répartition des week-ends de repos. 203
Figure 45 : Exemple de préférences individuelles : (*a*) Définition de l'horaire h_4 et, Comparaison avec les horaires h_1, h_2 et h_3, (*b*) selon la relation $\succcurlyeq_{L(wer),\, e}$ et (*c*) selon la relation $\succcurlyeq_{(L(wer),\, 6),\, e}$. 204
Figure 46 : Découpe du domaine réalisable du problème de pure répartition effectuée par la distance *VA* en *x*. 217
Figure 47 : Découpe de l'espace des critères ^{3} engendré par la contrainte souple de dominance de Pareto en ({mieux, incomparable}, *x*). 226
Figure 48 : Résolution Graphique de la contrainte de Lorenz en (surclasse, *x*) pour le problème de pure répartition sur 3 individus et une quantité totale de ressource égale à 100. 234
Figure 49 : Schéma de visualisation de l'établissement de la borne-consistance pour la contrainte de tri. 241
Figure 50: Exemple d'utilisation de la contrainte de tri et OGLO, pour établir un niveau de consistance pour la contrainte de Lorenz en (surclasse, *x*). 242
Figure 51 : Illustration de la contrainte SSE en (α, *x*) pour deux individus concernés *i* et *j*. 244
Figure 52 : Illustration des PB-contraintes souples étendues dont les solutions sont caractérisées par une permutation. 246
Figure 53 : Arborescence de recherche des couplages parfaits pour trois individus observés $I_e = \{i, j, k\}$. 251
Figure 54 : Interface utilisé pour paramétrer la résolution. 262
Figure 55 : Exemple de système de contraintes arc-consistant, mais pas globalement inverse consistant. 278

Abréviations et Indications de Lecture

❖ Abréviations

IA	Intelligence Artificielle (*Artificial Intelligence*)
MILP	Problème d'optimisation (Programme) linéaire en nombres mixtes (*Mixed Integer Linear Program*)
MINLP	Problème d'optimisation (Programme) non linéaire en nombres mixtes (*Mixed Integer Non Linear Program*)
MIP	Problème d'optimisation (Programme) en nombres mixtes (*Mixed Integer Program*) = MILP + MINLP
MOP	Programme mathématique multiobjectif (*Multi-Objective Program*)
PPC	Programmation par contraintes (*Constraint Programming*)
RO	Recherche Opérationnelle (*Operational Research*)
SIAD	Système interactif d'aide à la décision (*Decision Support System*)
TCS	Théorie du Choix Social (*Social Choice Theory*)

❖ Algorithmes

Les algorithmes sont décrits dans un pseudo-code admettant la notation mathématique usuelle. Les mots clefs, communs à tout langage impératif, sont écrits en gras comme :

si ($a = b$) **alors** **retourner** oui **fin si**

l'entête d'un algorithme est constitué du nom de l'algorithme en style '`courrier`', suivi des paramètres d'entrée mis entre parenthèses et séparés par des virgules, puis de deux points, ensuite du mot clé '**retourner**', et des arguments de sortie. Par exemple :

`AlgorithmeEgal`(a : rationnel, b : rationnel) : **retourner** booléen.

Lorsque l'algorithme est un processus de décision, les instructions renfermant un dialogue avec l'utilisateur sont surlignées en gris. Par exemple : Présenter(*Instance*). Si une réponse doit être retournée par l'utilisateur, la variable renseignée est suivie de l'instruction d'affectation (←), et la partie droite de l'affectation est surlignée en gris. Par exemple : réponse ← choisir(a, b) Sinon, les instructions purement calculatoires suivent la même description que les algorithmes classiques.

❖ Définitions

Les termes définis étant nombreux, ils sont introduits en gras dans le texte. Un index des mots est disponible en fin de document, indiquant en gras la page où ils sont définis. Lors de la définition, l'équivalent anglais est quelques fois ajouté en italique entre parenthèses quand il existe. Par exemple : La **théorie du choix social** (*social choice theory*) est ….

❖ Notations

Nous nous sommes efforcés d'adopter un système de notations uniforme pour plus de lisibilité. Ces notations sont introduites la première fois où elles sont nécessaires.

Des renvois fréquents permettent de mieux apprécier les liens existants entre les différentes parties de ce document. Nous donnons ci-après quelques notations usuelles :

$\mathcal{P}(E)$	ensemble des parties d'un ensemble E = ensemble des sous-ensembles de E.
card(E)	nombre d'éléments de l'ensemble E = cardinal de E.

Introduction

Documents essentiels pour la gestion des ressources humaines dans une entreprise, les plannings de travail de personnel nécessitent souvent une élaboration périodique et stratégique. Les raisons de cette élaboration sont multiples. Elles vont de la réduction des coûts de productivité à l'amélioration de la qualité de service aux clients, en passant par l'assurance d'une bonne qualité de vie aux employés.

Fortement combinatoire, ces problèmes ont tiré parti très tôt des avantages que procure l'ordinateur. Mais, dès le début, en plus des problèmes liés à la combinatoire, des difficultés à modéliser les préférences des différents acteurs sont apparues. La nature conflictuelle et incommensurable des préférences et points de vue des différents acteurs ajoute à la complexité de la tâche. Les mêmes difficultés sont apparues pour d'autres problèmes de décision et ont abouti à la naissance de disciplines nouvelles comme l'aide multicritère à la décision en recherche opérationnelle, la branche computo-symbolique de l'intelligence artificielle, mais aussi en sciences humaines et sociales avec notamment la théorie du choix social. Depuis les débuts de l'automatisation de cette tâche (années 50), l'information préférentielle servant à l'évaluation a pris différentes formes : (*a*) de la fonction objective unique appréhendant toutes les préférences pertinentes par une formule complexe, fondée sur une logique souvent obscure et peu propice à la communication, (*b*) aux procédures purement interactives entre le planificateur et le système informatique laissant toute initiative préférentielle à l'utilisateur.

Ces manières d'évaluer sont-elles compatibles ? L'essentiel des méthodes faisant un usage agrégé des préférences utilisent des techniques intuitives de conception, plus ou moins scientifiques ou critiquables. Elles manquent de fondements mathématiques solides. Peut-on faire autrement ?

L'objectif principal visé par ce travail est de *concevoir un système informatique dédié à l'élaboration de plannings de travail. Ceci en privilégiant le dialogue entre les différents protagonistes et en usant d'une représentation fiable de leurs préférences et de concepts de compromis forts, pouvant être partiels, comme l'efficacité et l'équité pour apporter une aide à la négociation en plus de l'aide combinatoire*.

Pour atteindre cet objectif, nous nous sommes d'abord intéressés à cette délicate question : **Comment évaluer la qualité des plannings ?** Un ensemble de questions viennent s'y greffer.

Tout d'abord, pourquoi évaluer les plannings ? Ne pourrait-on pas résoudre ces problèmes en utilisant des modèles exempts d'information de nature préférentielle ? En quoi la modélisation des préférences améliorerait-il la décision ? Les préférences sont généralement modélisées par les fonctions objectives associant à chaque planning une valeur. L'insatisfaction éprouvée dans la pratique est-elle due à l'inadéquation de ces modèles préférentiels à la nature même des préférences réelles ? Est-ce les méthodologies actuelles de construction des fonctions objectives qui sont inadéquates ? Ou est-ce que ce sont les deux ? Peut-on se baser sur une réelle méthodologie plus robuste et fiable ? Les problèmes d'élaboration de plannings de travail représentent les plannings de manière implicite (les solutions potentielles sont décrites par des contraintes). Cela a-t-il une influence sur la représentation des préférences ? Dans le cas positif, comment surmonter cet aspect combinatoire qui vient s'ajouter aux difficultés liées à la modélisation des préférences ? On parle souvent d'efficacité collective et d'équité comme caractéristiques indispensables à une bonne évaluation des plannings. Le premier concept est maintenant très bien connu en RO avec notamment la dominance de Pareto. Par con-

tre le second est limité à des spécialistes en sciences humaines. Ne peut-on pas bâtir des évaluations tenant compte uniquement de ces deux principes ? Quels rapports entretiennent la qualité de service aux clients et la qualité de vie des employés induites par les plannings avec ces deux concepts ?

Choix Majeurs

Afin d'apporter des réponses à toutes ces questions, nous avons suivi trois axes de recherche différents mais complémentaires : Le premier, la piste "*aide à la décision*" étudie les différentes formes que revêt une aide par un système informatique dans un processus de décision. Le second, la piste "*modélisation et représentation des préférences*" s'intéresse à décrire les préférences individuelles et collectives sous forme de modèles. Il s'intéresse également aux manières de les représenter en machine selon le contexte (solutions en extension, en compréhension, etc.). Et le troisième, la piste "*axiomatique et éthique*" fournit des fondements mathématiques solides aux procédures de décisions collectives, en identifiant les principes fondamentaux sous-jacents aux décisions et en les confrontant les uns aux autres. Certains peuvent être des principes moraux.

Cette thèse s'attache à donner une vue d'ensemble de chaque axe de recherche afin de mieux les combiner et les intégrer dans nos problèmes.

Le premier constat a été le manque de modèle générique combinatoire satisfaisant, tenant compte simultanément de ces trois aspects. Pour y remédier, ces travaux présentent un *couplage entre la programmation par contraintes* (PPC) *d'une part, et l'aide multicritère à la décision et le choix social d'autre part*. De cet union est né un nouveau *modèle combinatoire pouvant représenter une information préférentielle complexe pas forcément transitive et complète* : le *système de contraintes basé sur les préférences*. Nous définissons par la suite différents problèmes automatisables qui pourraient venir en aide aux utilisateurs pour aboutir à une décision réfléchie.

Dans ce cadre d'échange entre disciplines à des fins pragmatiques, *nous introduisons deux nouveaux axiomes de choix, valables dans un contexte général, mettant en avant le dialogue avec les utilisateurs en caractérisant des solutions de meilleure qualité, et en cas de conflit inévitable de fournir une description claire et non ambiguë des conflits incriminés*.

Le tout n'est pas de définir des modèles, il faut aussi développer des outils d'interfaces permettant la communication de cette information, pouvoir résoudre automatiquement certains problèmes pertinents associés en temps réel, etc.. Nous faisons un pas dans ce sens en *concevant une procédure arborescente résolvant un problème générique d'aide au choix, ainsi que des algorithmes de consistance*. Un prototype dédié à la planification infirmière interactive a été réalisé sur cette base d'algorithmes.

Du point de vue des concepts de compromis forts comme l'efficacité et l'équité, et de leur prise en compte dans les problèmes automatisables d'aide à la décision, des *algorithmes de consistance dédiés à certaines règles d'agrégation équitables et efficaces sont développés*.

Les Grandes Lignes du Plan de l'Étude.

Ce document se compose de deux parties complémentaires : une première partie consacrée à la modélisation et à l'aide à la décision (du second au sixième chapitre) ; Et une seconde partie dédiée au traitement algorithmique et aux tests (les septième et huitième chapitres).

Le premier chapitre présente d'une manière générale les problèmes d'élaboration de plannings de personnel, ainsi que leur intérêt pour les organisations. Nous argumentons après sur les avantages d'une automatisation.

Le second chapitre rappelle, dans les grandes lignes, les modèles théoriques utilisés pour modéliser les problèmes combinatoires (graphes, programmes en nombre mixtes, problèmes de satisfaction de contraintes). Ensuite, un état de l'art des modèles spécifiques à la planification d'horaires de travail est effectué. Les problèmes sont alors classés selon deux attributs principaux que sont la connaissance du nombre d'employés et la connaissance explicite ou implicite de l'ensemble des horaires individuels.

Nous proposons de replacer la conception de plannings de personnel dans une optique d'aide à la décision. De part sa nature sociale, elle devrait mettre en scène des outils de résolution propices à la négociation et au dialogue ainsi qu'à la recherche de solutions de compromis. Ainsi, le troisième chapitre se penche sur les notions de processus décisionnel et d'aide à la décision. Dans ce cadre, de nouvelles problématiques d'aide à la décision dédiées aux problèmes combinatoires sont décrites. Ensuite, nous nous penchons sur les différents modèles préférentiels de base apparus en théorie du choix social (TCS) et en aide multicritère à la décision (AMCD), afin de mieux représenter les préférences humaines et leurs interactions. Nous nous attardons plus particulièrement sur les *relations binaires de préférence* qui offrent plus de souplesse et sont mieux adaptés à l'aide à la décision que les modèles usuels fonctionnels d'évaluation de la qualité des plannings. Ce chapitre est clôturé par la présentation des axiomes de choix usuels que sont l'optimalité et la maximalité, suivie de la caractérisation de nouveaux axiomes de choix complémentaires, lorsque le modèle préférentiel est une relation binaire de préférence pas nécessairement transitive.

Les solutions de nos problèmes étant généralement évaluées à partir de plusieurs points de vue, émis par plusieurs acteurs (les employés, les gestionnaires, le service marketing, etc.), des modèles préférentiels adéquats sont présentés au quatrième chapitre. La règle d'agrégation est introduite comme outil de synthèse des préférences individuelles en préférences collectives. L'élaboration de plannings de travail de personnel peut être perçue comme un problème d'allocation de ressources. Ces problèmes ont donné lieu à des modèles préférentiels spécifiques en théorie du choix social et en éthique. Ils permettent de modéliser de manière particulièrement satisfaisante les problèmes de décision publique et de les évaluer en mettant en avant l'équité.

Les relations binaires de préférence sont définies dans un contexte où les solutions potentielles (dans notre cas les plannings réalisables) sont données en extension. Pour le moment, aucun modèle général n'a été conçu dans un contexte implicite. Par conséquent, le cinquième chapitre est ciblé sur la représentation efficace des modèles de préférences usant de la relation binaire de préférence dans un contexte implicite. Nous introduisons donc le terme de *système de contraintes basé sur les préférences*. Il permet de modéliser des problèmes combinatoires admettant une structure préférentielle complexe formée de l'avis de plusieurs personnes, synthétisé méthodiquement et hiérarchiquement en une relation binaire de préférence collective. Après, nous détaillons un ensemble de problèmes d'aide à la décision sur les systèmes de contraintes basés sur les préférences.

Le sixième chapitre est dédié aux différentes règles d'agrégation effectuant des jugements basés uniquement sur des principes fondamentaux caractérisant l'équité. Certaines de ces règles sont applicables que sur des problèmes de distribution de ressources, d'autres supposent uniquement la connaissance des préférences d'individus effectuant des jugements de comparaisons interpersonnelles. Ces règles d'agrégation sont alors utilisées pour définir une évaluation des plannings tenant compte aussi bien des coûts de production, des indicateurs de qualité

de service, que des préférences des employés pour élaborer un jugement sur les plannings qui soit à la fois efficace et équitable.

Les deux chapitres suivants sont consacrés aux algorithmes de résolution :

Le septième chapitre est consacré à la réalisation d'algorithmes de choix de solutions maximales et de consistance pour les systèmes de contraintes basés sur les préférences. Ainsi, les algorithmes de type recherche arborescente (Branch and Bound) ont été adaptés à ce nouveau modèle. En s'aidant de résultats en théorie polyédrale, nous caractérisons les programmes mathématiques multiobjectifs, dont les solutions Pareto-optimales correspondent aux solutions réalisables efficaces et équitables des problèmes d'allocation de ressources. Puis, nous confectionnons des algorithmes de consistance pour les règles d'agrégation équitables et efficaces. Dans ce cadre, nous explicitons le lien étroit entre la forme la plus générale des règles d'agrégation véhiculant des principes d'équité et d'efficacité, et les concepts de couplage et de couverture des sommets. Ce lien est après exploité pour concevoir un algorithme de filtrage adapté.

Un tel travail n'est pas complet si aucune mise en œuvre n'est effectuée. Par conséquent, un certain nombre d'algorithmes ont été programmés pour réaliser un prototype dans le cadre de l'élaboration de plannings infirmiers (chapitre 8).

Modélisation

et

Processus de Décision

Dans cette partie, nous argumentons sur le fait que la gestion des hommes nécessite des outils de modélisation spécifiques, ainsi qu'une méthodologie de résolution de problème basée sur le dialogue avec les protagonistes (élaboration préalable d'un processus de décision).

1. Problèmes d'Élaboration de Plannings de Travail de Personnel (PEPTP)

1.1. Généralités

1.1.1. Qu'est ce qu'un Planning ?

Les plannings sont des calendriers de travail, où figurent à la fois le temps, l'affectation du personnel, les jours et horaires de travail, et les congés et repos (HEUS [1996]). Certains plannings spécifient aussi les pauses et périodes de travail de la journée de chaque employé (plannings journaliers). D'autres répertorient uniquement les volumes horaires des semaines de chaque employé (plannings annuels). D'autres encore expriment en supplément un certain nombre de caractéristiques comme les tâches à effectuer en chaque période, les compétences nécessaires, etc.. L'intervalle de temps où un planning est élaboré est appelé l'**horizon** de planification.

Selon leur spécificité et les branches d'activités concernées, les plannings portent différents noms. Un planning spécifiant les programmes de travail de chaque employé nominativement sur un horizon d'un mois est appelé tableau de service. Lorsque le planning représente les programmes de travail et de repos non nominatifs sur un nombre entier de semaines, on parle de grille de travail.

Certains plannings sont cycliques, c'est-à-dire qu'ils reflètent une certaine périodicité des horaires individuels. Autrement, ils sont dits acycliques.

semaine	lundi	mardi	mercredi	jeudi	vendredi	samedi	dimanche
1	R	N	N	N	R	R	M
2	M	M	M	M	R	R	M
3	M	M	S	S	R	R	M
4	M	R	R	M	M	M	S
5	N	R	R	R	M	M	M
6	M	M	R	R	M	M	M
7	S	S	R	R	M	M	M
8	M	M	R	R	M	M	M
9	R	R	S	S	N	N	N
10	R	R	M	M	M	S	R
11	R	M	M	M	S	S	R
12	R	R	M	M	S	S	R

Légende :
R = repos
M = matin
S = soir
N = nuit

Tableau 1 : Exemple de grille de travail cyclique de périodicité 12 (PARTOUCHE [1998, Tab. 7.5]).

Comme document de travail au sein d'une entreprise, un planning revêt plusieurs formes qui dépendent de la précision souhaitée. Ainsi, certains plannings expriment la position des jours de travail et de repos (grille de repos), ou les étiquettes des jours de travail (matin, soir, nuit) et de repos (grille étiquetée), ou les vacations précisées pour les jours de travail (grille détaillée). (voir PARTOUCHE [1998]).

1.1.2. Unités et Acteurs

Une entreprise est un ensemble d'unités dialoguant entre elles, hiérarchisées (une unité peut, elle-même, se décomposer en sous-unités) et répartissant les salariés par fonction, par produits, par marché ou autres critères comme la localisation, etc. (MINTZBERG [1998]). L'élaboration de plannings peut être centralisée sur une unité fédératrice (comme l'entreprise toute entière ou une unité intermédiaire), ou effectuée indépendamment au niveau de la plus petite unité de l'entreprise (comme les unités fonctionnelles médicales : HEUS [1996], BONMATI [1998]). Cette unité est appelée aussi système cible ou autonome. Remarquons que l'autonomie du système cible, par rapport à la tache de planification des horaires, peut être partielle (système ouvert), des échanges limités d'agents inter-unités pouvant s'effectuer.

Les acteurs sont les personnes qui ont une influence sur le choix du planning.

Les employés, le corps managérial de l'unité et les clients sont les acteurs principaux. Ces trois types d'acteurs partagent une caractéristique fondamentale : si au moins l'une d'elles est déçue au point de partir, c'est tout le système qui s'arrête de fonctionner (MORIN [1999]).

Les pouvoirs publics, les syndicats et associations de consommateurs sont les acteurs secondaires, qui :

- représentent les différents acteurs principaux dans les processus de construction des règles de droit du travail, celles-ci s'exprimant sous forme de contraintes (durée maximale du temps de travail, nombre maximal de jours de travail successifs sans repos, etc.) ou de préférences (heures supplémentaires traduites par une majoration de salaire ou par des repos compensateurs (Lobry [1999]), durée légale du temps de travail, etc.) ;
- effectuent des contrôles (inspection du travail) pour veiller à l'application de règles.

1.1.3. A Quoi Sert un Planning ? Les Enjeux

Ce document de travail supporte une grosse partie de l'activité de gestion des ressources humaines. Depuis le début des années 80 en France, la gestion des ressources humaines a été reconnue comme une activité stratégique pour l'entreprise. Parler de « ressources humaines », c'est considérer non pas que les hommes <u>sont</u> des ressources, mais que les hommes <u>ont</u> des ressources (PERETTI [1998]). Ainsi, les premiers postes de DRH ont vu le jour, faisant des plannings un document de première importance pour un grand nombre d'entreprises. Avec cette reconnaissance, s'est vu accroître l'intérêt d'élaborer périodiquement des plannings de manière à être plus compétitif.

Prenons l'exemple des compagnies aériennes (RUSHMEIER *et al.* [1995]) : le planning complet final est utilisé à la fois pour être publié dans les systèmes de réservation de la clientèle un mois avant la réalisation, pour dicter les horaires des opérations de maintenance, des équipes de poste de pilotage (cockpit) et d'assistance (hôtesses et stewards), et de tout le personnel au sol.

Regardons un autre exemple avec les plannings effectués dans les unités hospitalières (SIFERD & BENTON [1992], WEIL *et al.* [1994]) : Une bonne qualité des soins doit être sans cesse respectée et un accompagnement des patients doit être effectué. La pénibilité psychologique de ce type de métier nécessite une vie équilibrée des infirmiers qui passe par des horaires proches de leurs attentes. Enfin, environs 70 % du budget d'une unité de soins est consommée par la masse salariale (BONMATI [1998]). Les plannings sont donc des documents essentiels pour ce type d'organisation. Leur élaboration relève du choix stratégique.

Les enjeux sont divers. L'élaboration des plannings permettent :

- aux entreprises exerçant une activité continue ou quasi-continue de répartir convenablement leur personnel. (compagnies aériennes, services aéroportuaires, entreprises de transport, hôpitaux, centres de renseignement ou réservation téléphonique, services de police, pompiers, etc.) ;
- aux entreprises cherchant à se rendre plus accessibles à la clientèle d'étaler les horaires d'ouverture. (restauration rapide, grands magasins, la distribution, les banques, etc..) ;

– à toutes les entreprises de surmonter leurs exigences de productivité ;
– aux entreprises de mieux gérer les présences et absences de leur personnel suite, par exemple, à des lois d'aménagement et de réduction du temps de travail (comme en France), ou aux différentes formes de contrats de travail (temps plein, temps partiel (80 %, 50 %, etc.), intérimaires, saisonniers, horaires individualisés, et bien d'autres) ;
– aux employés de mieux gérer leur temps libre (GAUVIN & JACOT [1999]). Un bon planning contribue à une bonne entente au sein de l'entreprise et un épanouissement au travail ;
– aux entreprises d'assurer une qualité de service minimale aux clients.

Les situations où un planning est utile sont nombreuses. Elles justifient l'existence de différentes formes de plannings dans un même système. Avant réalisation effective d'un planning, ce document est sans cesse modifié en fonction des nouvelles informations parvenues, concernant le très court, le court ou un plus long terme (PARTOUCHE [1998], CHAN [2002]). Les plannings sont en amont d'un ensemble d'activités de gestion routinières comme la paie, les bilans, etc. En d'autres termes, il s'agit de toute activité basée sur l'exploitation des données que renferment les plannings (*reporting*). Elle sert aussi à la planification des réunions ou des formations (HERTZ *et al.* [2000]).

1.2. Aspects Combinatoire et Négociation des Plannings

Les plannings sont nécessaires à la vie d'une entreprise. Leur élaboration n'est pas toujours automatique. Deux aspects de cette tâche concourent à faire usage de l'ordinateur pour améliorer son exécution : l'aspect combinatoire et l'aspect négociation. Ce sont les points que j'aborderai dans cette thèse.

1.2.1. Aspect Combinatoire

Pour un horizon de planification donné, du moment où les plannings ne sont pas explicitement connus, une dimension combinatoire fait son apparition.

Un planning correspond à l'affectation d'un horaire de travail par employé. Lorsque les horaires de travail admissibles pour chaque employé sont connus, et qu'il y a plus d'un horaire possible par personne pour l'horizon de planification visé, le nombre de plannings possibles peut exploser en nombre. Par exemple si notre personnel compte 10 employés, chacun pouvant travailler selon deux horaires différents, alors le nombre de plannings possibles s'élève à $2^{10} = 1024$ plannings !

La combinatoire est d'autant plus grande lorsque les horaires individuels ne sont pas explicitement connus.

Du moment où un planning doit être effectué sur un horizon dépassant 8 heures par jour, 35 heures par semaine[1] et/ou 5 jours sur 7, le nombre d'horaires parmi lesquels un choix est possible devient très grand. L'ordinateur est un candidat privilégié pour effectuer des traitements impliquant une telle masse d'information (générer un planning, comparer des plannings, etc.)

1.2.2. Quelques Organisations

Le besoin d'une gestion automatisée s'est fait sentir dans certains secteurs comme :
– le service autoroutier (agents de péage : DANTZIG [1954]),
– l'industrie automobile,
– le domaine du transport aérien (personnel naviguant (RUSHMEIER *et al.* [1995]), personnel au sol (PARTOUCHE [1998])), ferré et routier (compagnies de bus),

[1] Si les durées légales journalière et hebdomadaire de travail sont respectivement 8 et 35 heures.

- le domaine hospitalier (WARNER [1976], WEIL *et al*. [1994], HEUS [1996], JAUMARD *et al*. [1998]),
- les centres d'appels (SEGAL [1974]),
- le secteur bancaire et postal, la restauration rapide, le personnel hôtelier, le secteur pénitencier (CHAN & WEIL [2000]),
- le secteur de la distribution (caissiers), etc.

1.2.3. Comment Sont Évalués les Plannings ? : Aspect Négociation

Pour que les plannings élaborés soient satisfaisants, ils doivent vérifier un ensemble de contraintes et établir un meilleur compromis entre les préférences des différents acteurs.

Lorsque les différentes solutions alternatives sont connues, listées et en nombre restreint, et que le nombre d'acteurs est aussi restreint, une session de négociation standard se déroule de la manière suivante : Chaque acteur donne son opinion. Les points d'accord sont très vite expédiés et les points litigieux sont débattus. Et méthodiquement des solutions de compromis sont dégagées.

Les difficultés de négociation augmentent avec le nombre d'acteurs et le nombre de solutions alternatives. L'aspect combinatoire rend d'autant plus difficile la négociation, car les opinions sont plus difficiles à formuler. Les moyens informatiques apportent une aide certaine notamment dans l'acquisition et la confrontation des données préférentielles individuelles.

La Figure 1 résume les préférences des acteurs et les interactions et concepts de compromis liant ces préférences.

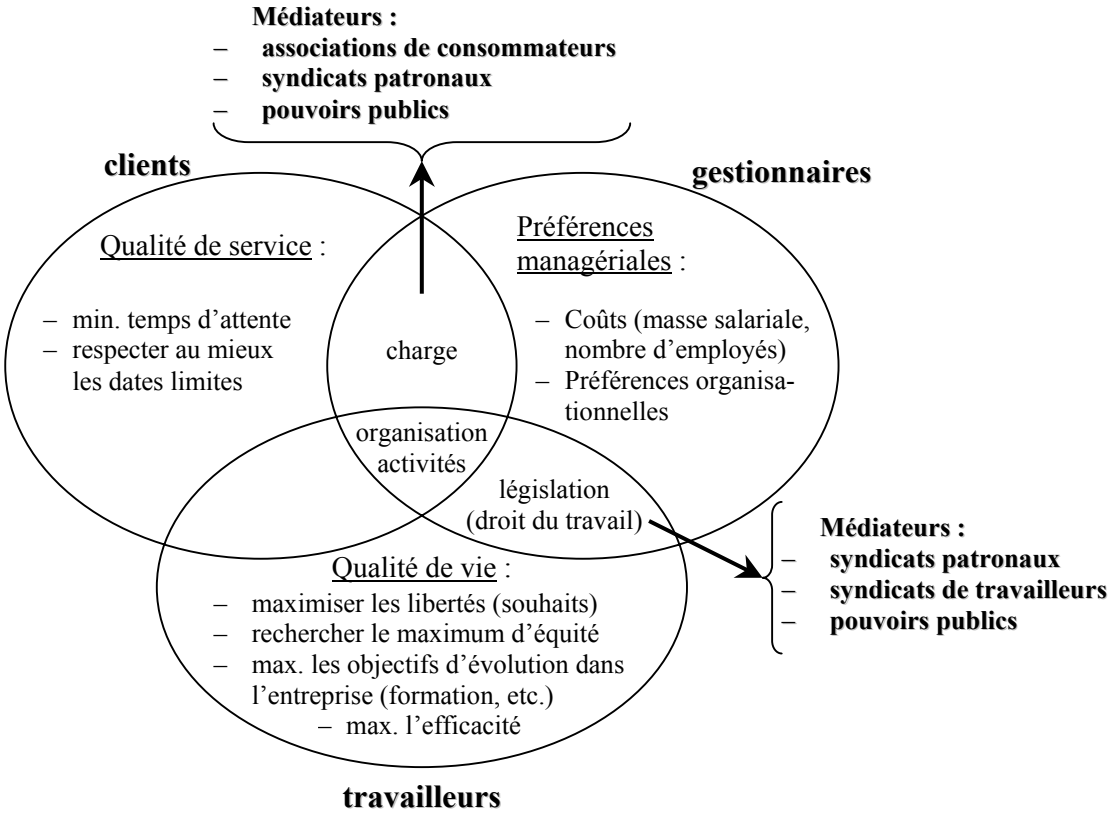


Figure 1 : Schéma général de l'environnement évaluationnel des plannings de travail.

Les préférences individuelles peuvent être agrégées selon différents critères pour évaluer leur qualité. Comme exemples de critères, nous avons la qualité de service aux clients induite par les plannings, qualité de vie des employés induite par les plannings et préférences managériales portant sur les plannings.

L'automatisation de l'élaboration des plannings peut être envisagée dès qu'un choix réfléchi s'impose entre plusieurs plannings. Pour le moment, nous parlons d'automatisation sans spécifier si elle doit être totale ou partielle.

1.2.4. Qui Peut se Charger de l'Élaboration ?

Cette tâche peut être centralisée ou déléguée à des cadres de l'unité appelés planificateurs. Précisons le rôle du planificateur dans cette tâche. Il représente :

- un décideur : il doit prendre la décision qui correspond le mieux aux préférences des différents acteurs concernés. Une certaine neutralité est recherchée (décision impartiale) ;
- un animateur, ou médiateur : il doit justifier son choix. Son pouvoir de décision est limité par ce compte rendu. Il occupe une place stratégique, à mi-chemin entre les différents acteurs ;
- un expert en planification : son expérience de la tâche fait de lui un interlocuteur privilégié pour évaluer rapidement et effectuer des jugements collectifs (il y a aussi le revers de la médaille : le favoritisme). Il est aussi l'une des personnes les plus aptes à juger de l'orientation à donner à la recherche de solutions de meilleure qualité (à se déplacer efficacement dans l'ensemble des solutions potentiellement candidates pour un choix pertinent) ;
- un utilisateur de logiciel de planification, et à ce titre, il peut être influencé, limité (options proposées inadéquates et restreintes) ou au contraire noyé (trop d'information : indécision) par les possibilités offertes par le protocole de résolution. Dans ces trois cas une « mauvaise » décision est prise.

Cependant, le planificateur n'est pas indispensable, car certains systèmes fonctionnent sans son intervention[2]. Le planificateur à lui tout seul ne connaît pas toutes les règles de bon sens liées à un individu particulier. Les règles de bon sens qu'il utilise sont génériques. En effet, il n'y a pas mieux que l'intéressé pour savoir s'il préfère travailler le matin après un long week-end.

Dans un cadre bien précis, cette tâche peut être réalisée par le personnel lui-même. On parle alors d'auto-planification (*self-scheduling*). Cette dernière possibilité n'est quasiment pas mentionnée dans la littérature RO, et est souvent survolée en IA. L'une des raisons possibles est l'aspect négociation mis en avant, cachant l'aspect combinatoire irréductible. Se pose alors la question de la (ou les) forme(s) que doit prendre une aide à la décision automatisée. De plus la formulation en termes de problème d'optimisation n'est pas évidente. C'est plutôt une planification, une élaboration **réactive**.[3]

1.3. Problématique Décisionnelle

La problématique décisionnelle de l'élaboration est la suivante :

L'élaboration de plannings de travail de personnel (PEPTP) *vise, pour un horizon de planification d'un jour à quelques mois, à dimensionner l'effectif et à aboutir à un programme définissant les horaires de travail et de repos de cet effectif. Ce programme devra respecter un ensemble de contraintes précises (contraintes de droit du travail et organisationnelles) et établir un meilleur compromis entre les préférences des différents acteurs.*

Cette problématique décisionnelle peut être nuancée selon les situations où les plannings sont utiles. Elle s'insère dans un processus de décision où les acteurs en décrivent l'aspect préférentiel.

---

[2] Certains écrits (MIETUS [1994], OLDENKAMP [1996]) lui accordent une importance qui, à mon avis, est à relativiser.

[3] Nous en reparlerons aux § 2.2.4.4 et 3.2.2.

2. Modèles Théoriques de Base et État de l'Art des Modèles en Élaboration de Plannings

« La modélisation est cette partie du processus de la recherche scientifique dans laquelle la réalité que nous cherchons à comprendre est traduite en un modèle qui sera utilisable dans les autres étapes du processus » extrait de CRÈTE & IMBEAU [1996].

Ce chapitre débute (§ 2.1.1) en effectuant quelques rappels sur la définition théorique d'un problème. Après s'être arrêté sur différentes relations liant les problèmes, nous introduisons les concepts généraux de la théorie de la complexité des problèmes solvables par les ordinateurs. Ensuite (§ 2.1.2 à 2.1.4), nous présentons les trois principaux modèles théoriques sur lesquels se basent les concepteurs pour élaborer les modèles spécifiques à l'élaboration des plannings : les systèmes d'inégalités linéaires, les graphes et les systèmes de contraintes. Au § 2.2 nous faisons un état de l'art des différents modèles utilisés pour les problèmes d'élaboration de plannings de travail de personnel (PEPTP), en notant lorsque cela est pertinent, les processus mis en œuvre pour acquérir cette information. Cet état de l'art classe en quatre groupes les PEPTP. Les classes sont reliées par des relations très souvent utilisées pour résoudre les problèmes. Nous clôturons ce chapitre (§ 2.3) en nous attardant sur quelques défauts des modèles usuels qui nous ont poussé à reformuler les PEPTP.

2.1. Modélisation Générale : Quelques Rappels

Dans cette section nous introduirons un certain nombre de notions et problèmes qui seront utilisés dans tout le reste du document.

Afin d'identifier les processus entièrement automatisables, une description des problèmes solvables par les ordinateurs est présentée ci-dessous, suivi des modèles théoriques génériques servant de référence aux modèles de la planification d'horaires de travail.

2.1.1. Diversité et Complexité des Problèmes

L'expérience montre que certains problèmes sont plus faciles que d'autres à résoudre. La théorie de la complexité des problèmes a été développée pour classer les problèmes par niveaux de difficulté, et décrire les problèmes qui peuvent être résolus de manière réaliste par les ordinateurs. Les outils mis à notre disposition par la théorie de la complexité sont devenus un standard dans l'évaluation pratique de la difficulté d'un problème.

Cette section, loin d'exposer toute la théorie de la complexité – pour une synthèse voir PAPADIMITRIOU [1994], GAREY & JOHNSON [1979], ou SAKAROVITCH [1984*b*, chap. 2] – présente certaines notions, dont il sera fait usage. Ainsi, après s'être attardé sur les problèmes, leur diversité et leurs liens – ces deux particularités étant très présentes lors des résolutions semi-automatiques –, différentes classes de problèmes, utiles pour la suite, seront présentées.

2.1.1.1. Les Problèmes

❖ La Notion de Problème

Un **problème** (*computational problem*) est conçu comme une relation $\Pi \subseteq \mathcal{I} \times \mathcal{S}$ entre les entrées ou **instances**, et les sorties ou **solutions**.

Pour qu'une instance d'un problème soit compréhensible par un ordinateur numérique, elle doit être décrite comme une séquence finie de symboles d'un ensemble fini arbitraire Σ appelé **alphabet**. De même, la solution d'une instance d'un tel problème est émise dans ce format. Généralement, $\mathcal{I}$ est infini alors que $\mathcal{S}$ peut être fini. L'ensemble des séquences finies de symboles de l'alphabet Σ est noté Σ^*.

❖ Résolution d'un Problème, Notion d'Algorithme

Nous disons qu'un ordinateur **résout** un problème $\Pi \subseteq \mathcal{I} \times \mathcal{S}$ s'il transforme n'importe quelle instance donnée $I \in \mathcal{I}$ en une solution $x \in \mathcal{S}$ satisfaisant $(I, x) \in \Pi$, si un tel x existe. Sinon retourne la solution "non". Il réalise cela en appliquant une séquence de transformations élémentaires prescrite par un **algorithme**[4].

Notons qu'une solution x satisfaisant $(I, x) \in \Pi$ est appelée **solution réalisable** pour l'instance I, sinon elle est dite **non réalisable** pour I. L'ensemble des solutions réalisables est noté $S_\Pi(I) \subseteq \mathcal{S}$. Lorsque l'information sur le problème à résoudre Π, est portée par l'instance, par exemple I_Π, nous noterons alors cet ensemble $S(I_\Pi)$ ou $S_\Pi(I)$ lorsqu'il sera nécessaire de distinguer des problèmes différents portant sur les mêmes instances.

Un **sous-problème** (ou un « cas spécial ») d'un problème $\Pi \subseteq \mathcal{I} \times \mathcal{S}$ est un autre problème $\Pi' \subseteq \mathcal{I}' \times \mathcal{S}'$ tel que $\mathcal{I}' \subseteq \mathcal{I}$ et $\Pi \subseteq \mathcal{S}' = \{ x \in \mathcal{S}$ tel que : $\exists\, I \in \mathcal{I}'$ satisfaisant $(I, x) \in \Pi\}$. En d'autres termes, Π' est un sous-problème de Π s'il répond à la même question que Π, mais seulement sur un sous-ensemble des instances de Π.

❖ Problèmes de Décision et Problèmes Combinatoires

Les problèmes peuvent être classés selon les propriétés de l'ensemble des solutions. Ainsi, un **problème de décision** vérifie que les solutions possibles sont réduites à la réponse "oui" ou la réponse "non". Dans le cas contraire (Lorsque la réponse est plus élaborée), on parle de **fonction problème**.

Quand, pour un problème Π donné, l'ensemble des solutions réalisables $S_\Pi(I)$ est fini, et ce pour toute instance $I \in \mathcal{I}$, on parle alors de **problème combinatoire**.

❖ Évaluation Usuelle des Solutions et Problèmes : l'Optimisation Simple

Dans une large majorité de problèmes, il est possible de séparer l'information véhiculée par une instance en deux groupes fonctionnels. Le premier décrivant l'ensemble des solutions alternatives, candidates au problème (on parle alors d'information sur la faisabilité du problème) ; et le second permettant de porter un jugement qualitatif sur les solutions (on parle d'information préférentielle). Les modèles sous-jacents à ces deux types d'informations sont appelés respectivement **modèle de faisabilité** et **modèle des préférences** du problème.

Dans les cas usuels, l'information préférentielle est modélisée par un couple (f_Π, op_Π) appelé **fonction objective** ou **vrai-critère**, avec :

- f_Π une fonction tributaire de l'instance I du problème Π considéré, et affectant à chaque solution candidate x un nombre rationnel $f_\Pi(I, x)$, appelé la **valeur** de x sachant I ; et
- op_Π une information booléenne donnant le sens de la préférence (optimisation), à savoir, la maximisation (resp. minimisation) lorsque op_Π = '≥' (resp. '≤').

[4] Bien entendu, certains problèmes ne sont pas calculables, c'est-à-dire qu'il n'existe pas d'algorithme permettant d'obtenir une solution à partir d'une instance. Pour plus d'information voir les références précitées.

Les solutions alternatives d'une instance I sont nommées **solutions réalisables** $S_\Pi(I)$ pour I, tandis que les solutions alternatives de meilleure qualité pour la fonction objective, sont appelées **solutions optimales** pour I. Plus formellement, une solution optimale pour une instance I est une solution candidate $x \in S_\Pi(I)$ telle que, pour tout $y \in S_\Pi(I)$[5],

$$f_\Pi(I, x) \ \mathsf{op}_\Pi \ f_\Pi(I, y).$$

L'ensemble des solutions optimales de Π est noté $B_\Pi(I)$. Remarquons que $B_\Pi(I) \subseteq S_\Pi(I)$. De plus s'il existe une solution réalisable, alors il existe une solution optimale. La réciproque est aussi vraie ($S_\Pi(I) \neq \emptyset \Leftrightarrow B_\Pi(I) \neq \emptyset$).

Le problème Π est un **problème d'optimisation simple** lorsque sa résolution pour une instance donnée est définie par la recherche d'une solution optimale pour la fonction objective associée.

Quand la fonction objective est « maximisante » (resp. « minimisante »), on parle de **problème de maximisation simple** (resp. **problème de minimisation simple**).

Remarque : L'information préférentielle peut être véhiculée par un autre modèle que le vrai-critère. En d'autres mots, l'identification d'un sous-ensemble non vide de $S_\Pi(I)$ possédant des propriétés qualitatives remarquables peut être réalisée, sans pour autant exhiber une fonction objective. Ces modèles préférentiels sont d'ailleurs l'objet du prochain chapitre.

❖ Description et Problèmes

Il est possible de créer une véritable nébuleuse de problèmes autour d'un seul autre, en se préoccupant d'énumérer toutes les solutions réalisables (resp. optimales), ou plus simplement en recherchant certaines propriétés de l'ensemble des solutions réalisables (resp. optimales). Il peut être fait usage, en pratique, des liens (différences et ressemblances) existants entre les problèmes, pour renseigner le décideur. Ainsi, après avoir présenté les définitions en compréhension et en extension d'un concept, nous introduirons quelques problèmes qui peuvent être dérivés d'autres.

❖ Représentations en Compréhension et en Extension

Il existe deux procédés fondamentaux pour décrire un concept. Lorsque celui-ci est décrit par la liste de ces manifestations (par exemple la liste de ces éléments pour le concept d'ensemble), on parle de **représentation explicite**. Dans le cas contraire, on parle de **représentation en compréhension** (ou représentation **implicite**). Par exemple, pour un problème donné, une instance fait une description en compréhension de l'ensemble des solutions réalisables.

❖ Problèmes de Dénombrement et d'Énumération

Continuons avec deux types de problèmes combinatoires : les **problèmes de dénombrement** (*enumeration* ou *counting problems* en anglais), et les **problèmes d'énumération** (*listing problems*). Ils sont tous les deux associés à un autre problème Π, le premier se préoccupant du nombre de solutions réalisables (noté $\#\Pi$) et le second de les restituer toutes (noté EnumΠ). Les formulations sont les suivantes :

$\#\Pi$: Etant donnée une instance I de Π, quelle est la cardinalité de $S_\Pi(I)$, c'est-à-dire, combien de solutions réalisables y a-t-il ?

EnumΠ : Etant donnée une instance I de Π, générer l'ensemble des solutions réalisables.

Les problèmes de dénombrement ne nous demandent pas de faire étalage de tous les éléments de $S_\Pi(I)$, comme c'est le cas du problème d'énumération, mais simplement que nous

[5] Nous notons « $f_\Pi(I, x) \ \mathsf{op}_\Pi \ f_\Pi(I, y)$ » pour dire : « $f_\Pi(I, x) \leq f_\Pi(I, y)$ lorsque op_Π = '≤' (resp. $f_\Pi(I, x) \geq f_\Pi(I, y)$ lorsque op_Π = '≥') »

déterminions combien il y en a. En termes de complexité, si nous avons un algorithme permettant de résoudre EnumΠ, alors le problème de dénombrement #Π et le problème initial Π sont résolus. La réciproque est par contre généralement fausse. Mais déjà à ce niveau, un premier classement partiel intuitif est possible.

❖ Description et Problème de Décision : Problèmes Partiels

Les problèmes de décision ont une importance capitale en complexité, du fait qu'à tout problème $\Pi \subseteq \mathcal{I} \times \mathcal{S}$, il peut-être associé un problème de décision dont la formulation est : « Etant donnée une instance I du problème Π, existe-t-il une solution réalisable ? », et appelé **problème de consistance simple** associé à Π (notation : consistance(Π)). En termes de complexité algorithmique, si nous avons un algorithme permettant de résoudre Π, alors le problème de consistance associé est résolu. La réciproque est généralement fausse, mais en classant les problèmes de décision nous obtenons un classement même partiel des autres.

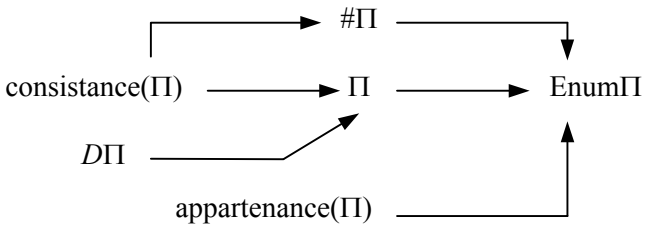
D'autres problèmes de décision peuvent être associés à un problème Π donné. Par exemple le **problème d'appartenance** (*membership problem*) associé à Π (notation : appartenance(Π)), dont la formulation est : « Étant donnée une instance I du problème Π, et un objet x quelconque de Σ^*, x est-il une solution réalisable de Π en I ? ($\Leftrightarrow x \in S_\Pi(I)$?) ».

Poursuivons avec les problèmes d'optimisation simple. Nous pouvons associer un problème de décision $D\Pi(B)$ au problème d'optimisation simple Π, en se donnant un paramètre rationnel B, et en questionnant sur l'existence d'une solution réalisable de valeur supérieure ou égale à B :

$D\Pi$: Etant donnée une instance I d'un problème d'optimisation Π, et un rationnel B, existe-t-il une solution réalisable $x \in S_\Pi(I)$ tel que $f_\Pi(I, x) \leq B$ lorsque op_Π = '≤' (resp. $f_\Pi(I, x) \leq B$ lorsque op_Π = '≥') ?

Bien sûr, si nous avons un algorithme résolvant le problème d'optimisation, nous en avons un résolvant le problème de décision associé. Par contre, la réciproque n'est pas toujours vraie (PAPADIMITRIOU [1994, chap. 17]).

On obtient en définitive la hiérarchie minimale suivante entre les variantes précédemment définies d'un problème.



Légende : Un arc d'un problème Π1 vers un autre Π2 signifie « un algorithme résolvant Π2 permet de résoudre Π1 ».
(les arcs transitifs n'ont pas été dessinés)

Figure 2 : Quelques liens de complexité entre problèmes.

Une variante d'un problème Π est appelée **problème partiel** de Π. Nous y reviendrons au § 3.2.2. Ces relations entre problèmes, données de manière plus ou moins intuitive, ont besoin d'assises théoriques et d'être généralisées. C'est la théorie de la complexité :

2.1.1.2. Quelques Classes de Problèmes

❖ Comment Mesurer la Qualité d'un Algorithme et d'un Problème ? Comment Prouver la Difficulté d'un Problème ?

La complexité d'un algorithme se mesure à partir de critères informant sur la quantité de ressources consommées, comme le temps et l'espace utilisés, en fonction de l'entrée. Ainsi, un **algorithme polynomial** en temps (resp. en espace) est un algorithme dont le temps

d'exécution (resp. l'espace occupé au cours de l'exécution) dans le pire cas est borné par une fonction polynômale en la taille de l'entrée. La complexité d'un problème se mesure par la complexité de l'algorithme le plus rapide qui le résout. Malheureusement pour un problème donné, la totalité des algorithmes pouvant le résoudre n'est pas connue.

Une autre solution est d'établir des relations entre les problèmes en rapport avec leur difficulté de résolution, d'où la notion de réduction de Turing polynomiale en temps.

❖ Transformation et Réduction Polynomiales entre Problèmes

Une **réduction de Turing polynomiale en temps** (ou simplement **Turing réduction**) d'un problème Π en un problème Π' est un algorithme A_Π qui résout Π en utilisant une hypothétique sous-routine $A_{\Pi'}$ résolvant Π' tel que :
si $A_{\Pi'}$ était un algorithme polynomial en temps pour Π', alors A_Π serait un algorithme polynomial en temps pour Π.
Dans ce cas on écrit $\Pi \propto_T \Pi'$, qui se lit « Π est Turing-réductible en Π' ». La relation $\propto_T$ est transitive.

Lorsque Π et Π' sont des problèmes de décision, on parle de **réduction polynomiale**. Si $A_{\Pi'}$ n'est appelé qu'une seule fois dans A_Π, et que Π et Π' sont des problèmes de décision, on parle alors de **transformation polynomiale**.

❖ Classes Fondamentales : NP, P, NP-complet, NP-Difficile

Face à la grande diversité des problèmes, et de manière à rendre la tâche moins difficile, les travaux de complexité se sont concentrés sur les problèmes de décision. Ceux-ci permettent d'attribuer une borne inférieure de complexité aux fonctions problèmes associés.

Une grande majorité des problèmes étudiés en pratique, formulés en problèmes de décision, appartient à la classe NP. La **classe NP** (*Non-deterministic Polynomial*) caractérise l'ensemble des problèmes de décision qui peuvent être résolus en un temps polynomial, par un algorithme non déterministe[6]. NP contient deux classes très importantes en pratique.
La première, la **classe P** (*Polynomial*) définit l'ensemble des problèmes dont le temps, dans le pire des cas, est polynomial en la taille de l'entrée. Cette classe modélise de manière plausible et mathématiquement productive, le concept empirique de « problème solvable pratiquement par un ordinateur ».
La seconde, la **classe des problèmes NP-complets**, identifie les problèmes de décision de NP les plus difficiles. Ils ont la propriété suivante : Un problème de décision Π de NP est NP-complet, si, pour tout autre problème Π' de NP, il existe une réduction polynomiale de Π' en Π. Ainsi, s'il existe un problème NP-complet pouvant être résolu par un algorithme polynomial, alors il existe un algorithme polynomial pour chaque problème de NP.
Il existe un nombre très important de problèmes NP-complets. Le premier des problèmes NP-complets, prouvé par Cook, est connu sous le nom de problème de **satisfaisabilité** (GAREY & JOHNSON [1979 § 2.6]).

Toute fonction problème $\Pi \subseteq \mathcal{I} \times \mathcal{S}$, dont le problème de consistance simple associé est dans la classe NP, vérifie :

- Son ensemble d'instances $\mathcal{I}$ peut être reconnu en un temps polynomial ;
- Etant donnée une instance $I \in \mathcal{I}$, son ensemble de solutions réalisables $S_\Pi(I) \subseteq \mathcal{S}$ vérifie que pour tout $x \in S_\Pi(I)$, la taille de x est polynomiale en la taille de I : étant donné une instance I et une solution x polynomiale en la taille de I, nous pouvons décider en un temps polynomial si $x \in S_\Pi(I)$.

[6] Algorithme qui comporte des instructions de choix qui devinent, guidant ainsi vers la bonne réponse. Remarquons qu'un algorithme polynomial ne comporte aucune instruction de ce type.

Un problème Π est **NP-difficile** s'il existe un problème de décision NP-complet Π' qui est Turing-réductible en Π. Ainsi, toute fonction problème, pour laquelle le problème de décision associé est NP-complet, est NP-difficile.

❖ Compléments : Problèmes de Dénombrement, #P-Complétude en Temps, Problèmes d'énumération, Sortie-Polynomialité

Des théories spécifiques ont été développées pour les problèmes de dénombrement et ceux d'énumération. Nous les renvoyons :

- pour le dénombrement, aux classes de complexité #P, #P-complet : GAREY & JOHNSON [1979 § 7.3] et PAPADIMITRIOU [1994], et
- pour l'énumération, aux classes de problèmes sortie-polynomaux en temps (*output-polynomial time*) : JOHNSON *et al.* [1988], EITER & GOTTLOB [1995], PAPADIMITRIOU [1997] et la notion d'algorithme adaptif dans Boisonnat & Yvinec [1995].

La suite de la section est consacrée à la présentation de trois modèles théoriques génériques que sont les systèmes d'inégalités linéaires mixtes, les graphes et les systèmes de contraintes. Bien sûr, bien d'autres structures permettent de modéliser des problèmes. Ces structures sont spécifiques à un domaine donné, et pour un objectif spécifié – i.e. offrent des facilités pour modéliser les systèmes de ce domaine – (par exemple les Réseaux de Pétri sont plutôt utilisés pour modéliser des processus de production, etc.). Les trois structures suivantes sont les plus utilisées en ordonnancement.

2.1.2. Graphes et Problèmes

❖ Graphe Orienté, Successeur, Prédécesseur, Chemin, Chaîne, Cycle, Circuit

Rappelons qu'un **graphe orienté** (BERGE [1983], SAKAROVITCH [1984*a*], BANG-JENSEN & GUTIN) $D = (V, A)$ est un ensemble fini non vide V d'éléments appelés **sommets** et un ensemble fini A d'éléments appelés **arcs**, tel que pour tout $e \in A$ est associé une paire ordonnée de sommets $(i(e), t(e)) \in V \times V$ appelés respectivement **sommet initial** et **sommet terminal** de e. On dit alors que $t(e)$ est un **successeur** de $i(e)$ et inversement $i(e)$ est un **prédécesseur** de $t(e)$. Lorsque qu'il existe des couples d'arcs $(e_1,e_2) \in A \times A$ tel que $i(e_1) = i(e_2)$ et $t(e_1) = t(e_2)$ nous parlons de graphe à arcs multiples. Dans le cas contraire, tout arc e est entièrement identifié par le couple ordonné de sommets $(i(e), t(e))$, et nous parlerons du couple (x, y) pour l'arc identifié par ce couple. Sauf cas spécifique et signalé, le terme de graphe orienté sera employé pour désigner un graphe sans arcs multiples, appelé aussi **graphe** orienté **simple**.

Pour un sommet $x \in V$ donné, un arc (x, x) est une **boucle**, et quand pour tout sommet, il existe une boucle, le graphe orienté D est dit **réflexif**. Lorsque pour tout arc $(x, y) \in A$ l'arc $(y, x) \in A$, alors D est dit **symétrique**. Si pour tout arc $(x, y) \in A$ l'arc $(y, x) \notin A$, alors D est dit **antisymétrique**. Lorsque pour tout triplet x, y, z de sommets tels que : $(x, y) \in A$ et $(y, z) \in A$ implique que $(x, z) \in A$, alors D est dit **transitif**. Si pour tout couple de sommets x, y, l'arc $(x, y) \in A$ ou l'arc $(y, x) \in A$, alors D est dit **complet**.

L'orientation peut ne véhiculer aucune information, l'important étant de connaître les deux sommets (appelés pour l'occasion **extrémités**) associés à un arc (appelé dans ce cas **arête**). La nouvelle structure obtenue est appelée **graphe non orienté**, et noté $G = (V, U)$.

Un graphe orienté (resp. non orienté) est souvent représenté graphiquement par des points correspondants aux sommets, et des flèches pointant sur le sommet terminal pour les arcs (resp. segments de courbe simple reliant les deux extrémités pour les arêtes).

Soit $D = (V, A)$ un graphe orienté (pas nécessairement simple) et $x, y \in V$. Un **chemin** (resp. une **chaîne**) de x à y dans D est une séquence d'arcs de D telle que :

- L'extrémité initiale (resp. l'une des extrémités) du premier arc de la séquence est x,
- L'extrémité terminale (resp. l'une des extrémités) du dernier arc de la séquence est y,
- L'extrémité initiale de chacun des autres arcs de la séquence coïncide avec l'extrémité terminale de l'arc précédent (resp. chaque arc intermédiaire de la séquence a une extrémité commune à l'arc précédent, et l'autre extrémité commune à l'arc suivant).

La définition d'une chaîne fait abstraction de l'orientation des arcs, et s'applique donc aux graphes non orientés.
Un **circuit** (resp. un **cycle**) est un chemin (resp. une chaîne) d'arcs <u>tous distincts</u> tels que x et y sont confondus ($x = y$).

❖ Forte Connexité, Graphe des Composantes Fortement Connexes

Soit $D = (V, A)$ un graphe orienté. Introduisons les deux relations binaires $\mathcal{C}_s$ et $\mathcal{C}_f$ – appelées respectivement **connexité** et **forte connexité** – définies sur l'ensemble V des sommets de D par :

(x, y) vérifie $\mathcal{C}_s$ s'il existe une chaîne entre x et y, ou si $x = y$;

(x, y) vérifie $\mathcal{C}_f$ s'il existe un circuit passant par x et y, ou si $x = y$.

La connexité (resp. la forte connexité) est une relation d'équivalence sur V, et le graphe $D / \mathcal{C}_s$ (resp. $D / \mathcal{C}_f$), dont les sommets sont les classes d'équivalence, est appelé **graphe des composantes connexes** (resp. **fortement connexes**).

❖ Fermeture Transitive d'un Graphe Orienté Simple

La **fermeture transitive** d'un graphe orienté simple $D = (V, A)$ est un graphe orienté simple $\tau(D) = (V, \hat{A})$ tel que :

$$(x, y) \in \hat{A} \Leftrightarrow \text{Il existe dans } D \text{ un chemin de } x \text{ à } y.$$

❖ Problèmes et Graphes : Sous-Ensembles Remarquables

Le concept de graphe orienté simple sur un ensemble fini V est équivalent à la notion de relation binaire sur V. Par conséquent, elle s'introduit de manière très naturelle comme support de modélisation d'un grand nombre de systèmes réels. Le fait que les graphes possèdent une représentation graphique claire fait d'eux une approche très féconde des relations binaires (soutient l'intuition et stimule l'imagination : SAKAROVITCH [1984*a*]).

Les problèmes sur les graphes sont très hétérogènes. Ils peuvent consister à identifier des sous-ensembles de sommets ou d'arcs non vides (cliques, couplages, etc.) possédant des propriétés remarquables données ; ou à enrichir les propriétés d'un type de graphe ; ou encore à étudier les transformations entre graphes ; etc.. Nous étudierons plusieurs de ces problèmes au § 3.4 et 7.4.5.

2.1.3. Les Systèmes d'Inégalités en Nombres Mixtes

<u>2.1.3.1. Les Programmes Linéaires en Nombres Mixtes</u>

Le **système d'inégalités linéaires en nombres mixtes** est l'un des modèles de base de la recherche opérationnelle et de l'optimisation mathématique. Il est défini par :

- une matrice $A \in \quad^{(m, n)}$, avec $A(i, \cdot)$ la i^{e} ligne de A,
- deux vecteurs $b_1 \in (\quad \cup \{-\infty\})^m$, $b_2 \in (\quad \cup \{+\infty\})^m$,
- un sous-ensemble $I \subseteq \{1, \ldots, n\}$, et
- les bornes des variables $l \in (\quad \cup \{-\infty\})^n$, $u \in (\quad \cup \{+\infty\})^n$, avec $l \leq u$;

caractérisant le sous-ensemble *MLS* de ^{n} suivant :

$$MLS(A, b_1, b_2, I, l, u) = \left\{ x \in \mathbb{R}^n \text{ tel que} \left| \begin{array}{l} b_1 \le Ax \le b_2 \\ l \le x \le u \\ x(i) \text{ entier, pour tout } i \in I \end{array} \right. \right\}$$

Le second type d'inégalités est appelé **contraintes de bornes**, et les contraintes astreignant certaines composantes du vecteur à être entières, sont appelées **contraintes d'intégrité**.
Lorsque $I = \emptyset$, *MLS* correspond géométriquement à un **polyèdre convexe** sur ^{n}, c'est-à-dire l'intersection d'un nombre fini de demi-espaces de ^{n}. Un polyèdre convexe borné est appelé un **polytope**. Un polytope *P* est aussi défini comme l'enveloppe convexe d'un nombre fini de vecteurs $\{x_1, \ldots, x_p\}$ de ^{n} appelés sommets du polytope :

$$P = \left\{ x \in \mathbb{R}^n \text{ tel que}: \exists \lambda \in \mathbb{R}^p \text{ avec } x = \sum_{i=1}^{p} \lambda(i) \cdot x_i,\ \lambda \ge 0 \text{ et } \sum_{i=1}^{p} \lambda(i) = 1 \right\}$$

Lorsque $I \neq \emptyset$, les éléments d'un système d'inégalités linéaires *MLS* sont les vecteurs du polyèdre convexe (obtenu en supprimant les contraintes d'intégrité de *MLS*) pour lesquels les composantes de *I* sont entières.
Ces objets géométriques ont été étudiés en profondeur en théorie polyédrale. On ne fera pas de synthèse ici, le lecteur intéressé étant invité à lire par exemple NEMHAUSER & WOLSEY [1988], SCHRIJVER [1986] ou ZIEGLER [1995].

Un **problème d'optimisation linéaire en nombres mixtes** (ou programme linéaire en nombres mixtes ; MILP en abrégé) est un problème d'optimisation sur les systèmes d'inégalités linéaires en nombres mixtes. Sa description est la suivante :

MILP : Étant donnés un vecteur $c \in$ ^{n} et un système d'inégalités linéaires en nombres mixtes *MLS* sur ^{n}, trouver un vecteur $x^* \in MLS$ vérifiant $c^T x^* = \max \{c^T x \text{ tel que } x \in MLS\}$ lorsque *MLS* est non vide et qu'une telle solution est finie[7] ; sinon retourner "non".

Le vecteur *c* définit la fonction objective. Si toutes les variables sont entières, i.e., $I = \{1, \ldots, n\}$, le programme linéaire en nombres mixtes est appelé **programme linéaire en nombres entiers** (ILP). Si les variables de *I* sont astreintes à être 0 ou 1, i.e., $l_i = 0$ et $u_i = 1$ pour tout $i \in I$, le problème d'optimisation est appelé **programme linéaire mixte en 0/1**. Lorsque toutes les variables sont réelles, i.e., $I = \emptyset$, on parle de **programme linéaire** (LP).

Remarquons que lorsqu'une instance de LP admet des solutions optimales (le polyèdre des solutions réalisables est non vide, et les solutions optimales ne se trouvent pas à l'infini), alors au moins un sommet du polyèdre est une solution optimale. Par conséquent, la recherche d'une solution optimale de LP peut se limiter aux sommets du polyèdre. Cette relation entre le continu (l'ensemble des solutions et celui des solutions réalisables sont des ensembles continus) et la combinatoire (le nombre de sommets est fini) a été très exploitée dans la pratique et a donné naissance à la fameuse méthode de résolution « simplexe ».
Pour les MILPs, cette information n'est plus valable, sauf dans des cas très particuliers, où les sommets du polyèdre – obtenu en supprimant les contraintes d'intégrité – ont les composantes indicées par *I* entières (flots, propriété de totale unimodularité et de total dual intégrité de la matrice *A*, etc.).
La dichotomie sur les variables (entières et réelles), traduit en réalité une différence de complexité. Le problème de décision associé à LP est polynomial, alors que ceux associés à ILP et MILP sont NP-complets dans le cas général (SCHRIJVER [1986]).

[7] Lorsque l'ensemble des solutions réalisables est non vide, il peut n'exister aucune solution optimale dans ^{n}, ceux-ci se trouvant à l'infini.

2.1.3.2. Modélisation : Formulations, Opérateurs Logiques et Extension Non Linéaire

❖ Les différentes formulations

Il est possible de représenter un problème réel de différentes façons par les MILPs. Cela est dû au processus même de modélisation. Un modèle n'étant qu'une représentation du monde, les informations pertinentes définissant le modèle n'est pas unique. On en reparlera régulièrement dans ce document.

❖ Opérateurs logiques sur les systèmes d'inégalités linéaires

Des outils méthodologiques existent pour aider à modéliser un système cible en MILP. Il est ainsi d'usage de faire intervenir des variables de décision bivalentes, prenant la valeur 1 lorsqu'une proposition donnée est vérifiée, et 0 sinon. D'autres artifices sont aussi utilisés comme, l'approximation, l'opérateur logique « ou », etc. (voir NEMHAUSER & WOLSEY [1988])[8].

❖ Les Programmes Non Linéaires en Nombres Mixtes

Pour les besoins spécifiques aux systèmes étudiés, des modèles plus généraux et/ou nuancés sont aussi utilisables, notamment les **programmes non linéaires en nombres mixtes** (MINLP en abrégé). Ces problèmes admettent pour ensemble de solutions réalisables, les **systèmes d'inégalités non linéaires en nombres mixtes** définis par :

- les fonctions h_i avec $i \in \{1, \ldots, m_1\}$ et hh_j avec $j \in \{1, \ldots, m_2\}$ de ^{n} dans , et
- un sous-ensemble $I \subseteq \{1, \ldots, n\}$;

caractérisant le sous-ensemble *MNLS* de ^{n} suivant :

$$MNLS(h, hh, I) = \left\{ x \in \mathbb{R}^n \text{ tel que} \left| \begin{array}{ll} h_i(x) = 0 & \text{pour tout } i = 1, \cdots, m_1 \\ hh_j(x) \leq 0 & \text{pour tout } j = 1, \cdots, m_2 \\ x(i) \text{ entier,} & \text{pour tout } i \in I \end{array} \right. \right\}$$

Un MINLP est alors décrit comme suit :

MINLP : Etant donnés une fonction objective f de ^{n} dans , et un système d'inégalités non linéaires en nombres mixtes *MNLS*, trouver un vecteur $x^* \in MNLS$ vérifiant[9] :

$$f(x^*) = \max \{f(x) \text{ tel que } x \in MNLS\}$$

lorsque *MNLS* est non vide et qu'une telle solution est finie[7] ; sinon retourner "non".

Les MINLPs sont principalement utilisés en ingénierie chimique[10]. Lorsqu'il n'y a pas de variables entières, i.e. $I = \varnothing$, on parle de **programme non linéaire** (NLP). Les programmes non linéaires ont été très utilisés en économie notamment. Et il existe un grand nombre d'algorithmes de résolution.

❖ Programmes Mathématiques Multiobjectifs

---

[8] Faisons quand même un clin d'œil à la programmation disjonctive laquelle se préoccupe de résoudre des problèmes sur des systèmes formés de disjonctions de systèmes d'inégalités linéaires en nombres mixtes (BALAS [1979]).

[9] Les systèmes de nombres de l'ordinateur étant discrets et finis, cette condition est souvent formulée en tenant compte d'une erreur ε (avec $0 < \varepsilon << 1$) et d'une distance dans ^{n}. Le but est alors de rechercher une solution réalisable approchant x^* à une distance ε près.

[10] Pour des approfondissements, voir FLOUDAS [1995].

Lorsqu'un programme en nombres mixtes admet plusieurs fonctions objectives $f_1, \ldots, f_q$, on parle de **programme mathématique multiobjectif** (MOP). Les fonctions objectives permettent de modéliser plusieurs points de vue. Le MOP n'est pas clairement défini, car aucune indication n'est donnée sur la manière d'utiliser les fonctions objectives pour identifier une solution de meilleure qualité. Par conséquent, ce terme est généralement utilisé pour désigner tous les problèmes bien définis associés.

❖ Conclusion

En conclusion, nous dirons que, ces modèles théoriques génériques reposent sur les axiomes suivants :

- Toute variable est à valeur dans ou , ou tout autre ensemble possédant au moins leurs propriétés structurelles (algébriques, topologiques et ordinales) ;
- Toute contrainte a une forme analytique (fonctionnelle, linéaire) ou peut être convenablement représentée par un système d'inégalités fonctionnelles (linéaires) ; et
- Le modèle d'évaluation des solutions peut être représenté par un vrai-critère ou fonction objective.

La majorité des problèmes auxquels nous sommes confrontés, admettent ces axiomes comme absolument nécessaires à leur définition.

Ces problèmes ont été abondamment traités en RO et optimisation mathématique ces 50 dernières années. Par conséquent, une quantité impressionnante d'outils algorithmiques et logiciels efficaces sont disponibles.

Dans la prochaine section, une dernière structure est présentée qui, ces dernières années, a remporté beaucoup de succès dans la modélisation de systèmes réels complexes : les Systèmes de Contraintes.

2.1.4. Les Problèmes sur les Systèmes de Contraintes

Apparu sous sa forme actuelle (générale), dans les années soixante dix (MONTANARI [1974], MACKWORTH [1977], FREUDER [1978]), le **système de contraintes** (appelé aussi Réseau de Contraintes ou Système Contraint dans la littérature) se définit par la connaissance d'un ensemble fini de variables, un domaine de valeurs par variable, et un ensemble fini de relations, appelées contraintes, que doivent vérifier les valeurs prises par les variables.

Depuis son apparition, ce modèle générique a donné lieu à d'innombrables études et applications industrielles, allant de problèmes en vision, robotique, ordonnancement, conception de circuits VLSI, planification, etc.[11]. Ainsi, plusieurs outils logiciels génériques basés sur cette technologie ont vu le jour, comme `IlogSolver` de la société ILOG [1995], `Chip` de COSYTEC [1995], etc.

Après avoir donné une définition formelle du système de contraintes, nous présenterons les Problèmes de Satisfaction de Contraintes et Problèmes d'Optimisation sur les Contraintes, qui font un usage central de ce modèle générique. Les deux dernières sections sont consacrées à la façon dont l'information véhiculée par les domaines et les contraintes est restituée en pratique. Cette information supplémentaire aboutit à une spécialisation (dans le cas des domaines) et à une généralisation (en terme de problème dans le cas des contraintes) de la définition du système de contraintes. Nous présentons ensuite quelques résultats de complexité issus de la littérature.

2.1.4.1. Systèmes de Contraintes et Problèmes de Satisfaction de Contraintes

Dans cette section, nous introduisons un certain nombre de notions.

[11] Voir ROSSI [1999] pour une synthèse des logiciels généraux et applications industrielles.

❖ Définitions

Un **Système de Contraintes** $\mathcal{R} = (V, D, C)$ est défini par la donnée d'un ensemble fini $V = \{v_1, \ldots, v_n\}$ de variables en extension, un ensemble fini D de domaines associant à chaque variable $v \in V$ un domaine de valeurs $D(v)$, et un ensemble fini C de contraintes.

Chaque contrainte est définie sur un sous-ensemble ordonné de variables et impose des restrictions sur la combinaison des valeurs que les variables de ce sous-ensemble peuvent prendre.

Plus formellement, une **contrainte** c de C, est un couple $(V(c), S(c))$ où :

– la première composante $V(c)$ est un sous-ensemble ordonné de V, et
– la seconde composante $S(c)$ est un sous-ensemble de $\bigtimes_{v \in V(c)} D(v)$, appelée **sémantique** de c

Le nombre de variables de l'ensemble $V(c)$ est appelé l'**arité** de la contrainte c. Nous noterons le produit cartésien d'un ensemble de variables V' quelconque par $\mathcal{D}_{V'} = \bigtimes_{v \in V'} D(v)$. Un élément x du produit cartésien $\mathcal{D}_V = D(v_1) \times \ldots \times D(v_n)$ est appelé une **solution** du système de contraintes $\mathcal{R}$. Soit un ensemble de variables $V' \subseteq V$. La **restriction** $x_{|V'}$ d'une solution x aux variables de V' est l'élément de $\mathcal{D}_{V'}$ tel que $x_{|V'}(v) = x(v)$ pour tout $v \in V'$. La solution x de $\mathcal{R}$, **satisfait** la contrainte c de C si la restriction de x aux variables de $V(c)$ est un élément de la relation $S(c)$ ($\Leftrightarrow x_{|V(c)} \in S(c)$). On dit aussi que x est **consistante** avec c. Dans le cas contraire on dit que x est **inconsistante** avec c. Nous arrivons à la définition d'un problème de satisfaction de contraintes :

Problème de satisfaction de contraintes (CSP en abrégé, *Constraint Satisfaction Problem*) :
Etant donné un système de contraintes $\mathcal{R} = (V, D, C)$, trouver une solution x de $\mathcal{R}$ qui satisfait <u>simultanément</u> toutes les contraintes de C, si un tel élément existe, sinon retourner "non".

L'ensemble des solutions réalisables d'un CSP pour l'instance $\mathcal{R}$ est noté $S_{\text{CSP}}(\mathcal{R})$ au lieu de $S(I_{CSP})$ avec $I_{CSP} = \mathcal{R}$. Dans la communauté IA, une solution est appelée une **instanciation**, alors qu'une solution <u>réalisable</u> est appelée **instanciation consistante**.

Remarquons qu'aucune restriction cardinale (ensemble fini, ensemble dénombrable, etc.) ou structurelle (algébrique, topologique, etc.) n'est imposée aux domaines de variables, comme c'est le cas des systèmes d'égalités et inégalités en nombres mixtes. Par conséquent, n'importe quel ensemble structuré ou non, peut faire office de domaine. La définition d'une contrainte, bien entendu, peut tirer partie des propriétés structurelles cardinales et ordinales des différents domaines de variables sur lesquelles elle porte.

Les systèmes de contraintes peuvent être conçus comme un élargissement du modèle linéaire par affaiblissement des hypothèses structurelles des domaines et contraintes. A ce niveau, cette généralisation correspond à une relaxation axiomatique des systèmes d'inégalités linéaires[12].

La définition usuelle présentée ci-dessus d'une contrainte et d'un domaine ne donne aucune précision sur la manière dont ils sont codés. Mais, pour une utilisation algorithmique, il est nécessaire d'aller plus loin. Nous y consacrerons les deux sections suivantes.

<u>2.1.4.2. Sur la Modélisation des Domaines</u>

❖ Différents Codages

[12] Le schéma inverse – approcher tout système de contraintes par un système d'inégalités linéaires en nombres mixtes – a été étudié dans la littérature. Voir par exemple NIELSEN *et al.* [2000].

La variété des problèmes à résoudre, a suscité l'utilisation de différents modèles pour représenter un domaine de variables. Historiquement, c'est l'ensemble de valeurs en extension qui a été adopté dans les définitions :

$$D(v) = \{d_1(v), \ldots, d_{q(v)}(v)\}$$

Mais, sa validité est restreinte aux domaines finis et très petits. Les CSP faisant un usage exclusif de systèmes de contraintes à domaines finis en extension sont appelés CSP à domaines finis en extension. Ce sont les plus utilisés dans la pratique et les plus étudiés en théorie.

Cependant certains problèmes exigent des domaines modélisés en compréhension. L'utilisation des intervalles a alors été entreprise.

Ainsi, étant donné un ensemble totalement ordonné $(B, \geq)$ – par exemple l'ensemble des entiers, des rationnels ou des réels – un **intervalle** X dans B est l'ensemble de tous les éléments de B compris entre deux éléments donnés a et b de B :

$$X = \{e \in B \text{ tel que } a \leq e \leq b\} \text{ noté } [a, b]. \text{ Et, dans le cas où a} > b, X = \varnothing.$$

Les systèmes de contraintes dont les domaines sont uniquement des intervalles sont envisagés par plusieurs auteurs (par exemple BENHAMOU & OLDER [1997], JAULIN *et al.* [2001]. D'autres références seront données à la section 7.4.3), et les CSP qui en font usage sont appelés **problèmes de satisfaction de contraintes à intervalles** ou ICSP.

Un tel codage permet de réduire considérablement la représentation d'un intervalle d'entiers par exemple, en passant d'une représentation explicite de taille en $O(\,(b - a)\cdot\log(\max\{a, b\})\,)$, à une représentation implicite de taille en $O(\,\log(\max\{a, b\})\,)$. Sans oublier les possibilités de modélisation d'ensembles non discrets impossibles à représenter de manière explicite[13]. D'un point de vue complexité algorithmique, un algorithme recevant en donnée un domaine $D(v)$ formé des entiers entre a et b, et s'exécutant en un temps maximum de $O(P(a, b))$, avec P un polynôme, est polynomial en la taille de $D(v)$ donné en extension, mais par contre exponentiel en la taille de $D(v)$ donné par un intervalle en compréhension. Nous pouvons poursuivre nos remarques sur la complexité : Le nombre de solutions d'un problème combinatoire séparable de NP est généralement utilisé comme borne supérieure de sa complexité pratique ; Par définition, elle est donnée par le produit du cardinal des domaines :

Taille de l'espace de recherche des solutions réalisables du CSP pour le système de contraintes $\mathcal{K} = (V, D, C)$ à domaines en extension :

$$\prod_{v \in V} \mathrm{card}(D(v)). \tag{2–1}$$

Cette borne supérieure correspond à l'algorithme qui consiste à générer toutes les solutions, puis à tester la satisfaction de chaque contrainte[14].

Mais lorsqu'au moins un des domaines est implicite, la complexité ne peut plus être mesurée de cette manière. C'est le cas de ILP, LP et MILP (voir SCHRIJVER [1986]). Il est alors nécessaire de montrer l'existence d'une réduction polynomiale d'un problème dont la complexité est connue, vers notre problème (sous entendu, le problème de décision associé) pour identifier une borne supérieure de sa complexité (conséquence du théorème 2.1 dans GAREY & JOHNSON [1979]).

Ces dernières années, des travaux ont été entrepris vers une généralisation des intervalles (HICKEY *et al.* [2001], HICKEY [2001]), afin de tenir compte de l'inclusion ou non des bornes de l'intervalle dans le domaine. Ainsi, un **intervalle généralisé** est défini par un quadruplet $(a, b, \alpha, \beta) \in B \times B \times \{\text{non, oui}\} \times \{\text{non, oui}\}$, de la manière suivante :

---

[13] Notons aussi les possibilités qu'offre ce type de systèmes à intervalles pour étudier des problèmes d'incertitude, par l'utilisation d'une approche ensembliste au lieu d'une approche probabiliste. Voir sur ce point les travaux de JAULIN *et al.* [2001].

[14] Dans le cadre des CSPs, on suppose alors que pour toute contrainte, l'algorithme permettant de répondre si une solution satisfait ou non cette contrainte, est polynomial et peut donc être compté comme une opération élémentaire. Dans le cas contraire, le problème n'est plus dans NP.

$D(v) = \{e \in B$ tel que $a < e < b$ ou $e = a$ si $\alpha =$ oui ou $e = b$ si $\beta =$ oui$\}$ noté $[a, b]_{\alpha, \beta}$.
Et, dans le cas où $a > b$ ou [$a = b$ et ($\alpha =$ non ou $\beta =$ non)], nous avons $D(v) = \varnothing$.

La généralisation à toute **union finie d'intervalles disjoints** de B a été aussi considérée (HICKEY *et al*. [2001], HICKEY [2001]) :

$$D(v) = \bigcup_{k=1}^{m(v)} [a_{v,k}, b_{v,k}]_{\alpha_{v,k}, \beta_{v,k}} ,$$

avec $m(v) > 0$ le nombre d'intervalles disjoints non vides formant $D(v)$.

En continuant dans cette voie, rien ne nous empêche de considérer des domaines implicites autres, où :
- chaque élément peut être obtenu via un algorithme,
- il est possible d'effectuer des opérations d'ajout et de suppression d'éléments, sans incidence notable sur la complexité, et stable pour le type de système de contraintes utilisé.

❖ Opérations Ensemblistes sur les Domaines

Quel que soit le codage utilisé pour représenter le domaine des valeurs d'une variable, une variable de type domaine possède un ensemble d'opérations minimales nécessaires tels que :
- l'appartenance : $d \in D(v)$?,
- l'affectation : $D(v) \leftarrow E$, avec E un ensemble quelconque, compatible avec le codage,
- la suppression,
- l'ajout d'éléments (intersection et union d'ensembles compatibles).

❖ Opérations Arithmétiques sur les Domaines

Lorsque B est structuré algébriquement, une arithmétique des intervalles a été développée (voir BENHAMOU & OLDER [1997]). Même chose pour les unions finies d'intervalles disjoints, mais des problèmes en termes de complexité spatiale viennent limiter son utilisation (nous en reparlons au § 7.4.3.4)[15].

Face au gain indéniable (compactage de l'information) obtenu par l'utilisation de représentations implicites, il y a un revers : les opérations considérées comme élémentaires pour une représentation, peuvent s'avérer difficiles pour une autre.

Lorsque le codage d'un domaine est différent du cas fini en extension, il est nécessaire :
- de refaire l'inventaire des opérations qui peuvent être considérées comme élémentaires : des opérations ensemblistes (comme l'union et l'intersection), aux opérations arithmétiques ;

et quelques fois,
- d'adapter les définitions pour conserver leur stabilité. Par exemple l'union, ou encore la division sur l'ensemble des intervalles n'est pas stable (n'est pas toujours un intervalle). Il est nécessaire d'adapter sa définition pour conserver la stabilité, ou considérer un codage plus étendu. Cette dernière possibilité a donné lieu à la création des unions finies d'intervalles disjoints pour résoudre le problème de stabilité de la division et de l'union.

En résumé, nos ensembles de valeurs peuvent ne pas être donnés en extension. Et de réelles avancées peuvent être effectuées avec l'utilisation d'ensembles de valeurs données en compréhension.

[15] Des structures intermédiaires entre l'intervalle et l'union d'intervalles – discrétisation des domaines – ont été proposés pour palier à cette explosion de complexité. Sur ce point voir par exemple SAM-HAROUD & FALTINGS [1996].

2.1.4.3. Sur la Modélisation des Contraintes

Les contraintes sont, dans un grand nombre d'articles théoriques, fournies par la liste des combinaisons satisfaisant la contrainte. Mais, pour qu'ils soient applicables industriellement (les problèmes d'ordonnancement par exemple), ces modèles ont évolué en intégrant d'autres paradigmes de modélisation.

Tout d'abord, conséquence des domaines implicites, la sémantique $S(c)$ d'une contrainte c peut aussi être donnée en compréhension. C'est le cas des contraintes se formulant analytiquement.

Un autre point, la récurrence :

❖ Opérateurs Logiques Usuels sur les Contraintes

Il est possible de définir les trois opérateurs logiques usuels sur l'ensemble des contraintes :

– L'opérateur binaire de **conjonction** « et » : pour toute contrainte c_1 et c_2, leur conjonction $c_3 = c_1$ et c_2 est la contrainte définie comme suit :

$$V(c_3) = V(c_1) \cup V(c_2) \text{ et}$$

$$S(c_3) = \left\{ x \in \bigtimes_{v \in V(c_1) \cup V(c_2)} D(v), \text{ tel que : } x_{|V(c_1)} \in S(c_1) \text{ et } x_{|V(c_2)} \in S(c_2) \right\}$$

– L'opérateur binaire de **disjonction** « ou » : pour toute contrainte c_1 et c_2, leur disjonction $c_3 = c_1$ ou c_2 est la contrainte définie comme suit :

$$V(c_3) = V(c_1) \cup V(c_2) \text{ et}$$

$$S(c_3) = \left\{ x \in \bigtimes_{v \in V(c_1) \cup V(c_2)} D(v), \text{ tel que : } x_{|V(c_1)} \in S(c_1) \text{ ou } x_{|V(c_2)} \in S(c_2) \right\}$$

– L'opérateur unaire de **négation** « non » : pour toute contrainte c_1, sa négation c_2 = non[c_1] est la contrainte définie comme suit :

$$V(c_2) = V(c_1) \text{ et } S(c_2) = \left\{ x \in \bigtimes_{v \in V(c_1)} D(v), \text{ tel que : } x_{|V(c_1)} \notin S(c_1) \right\}$$

Remarquons que ces définitions permettent de lier les opérateurs logiques sur les contraintes et les opérateurs ensemblistes sur les sous-ensembles de produit cartésien de domaines.

Le processus qui permet, à partir d'un ensemble de contraintes, d'obtenir d'autres contraintes en appliquant des opérateurs, est appelé **processus récursif**.

Le processus récursif est très utilisé en modélisation dans la pratique. Lorsqu'une contrainte est difficile à définir, nous essayons de la décomposer en contraintes plus simples reliés par des opérateurs. C'est ainsi qu'en partant d'un ensemble de contraintes primitives, d'un certain nombre d'opérateurs, et en appliquant un nombre fini de fois les opérateurs, il est possible de créer une assez grande variété de contraintes évoluées, complexes. C'est sur ces bases récursives que la programmation logique par contraintes a été développée. Pour une présentation de cette théorie, voir VAN HENTENRYCK [1989], FAGES [1996]. D'autres paradigmes de programmation comme l'objet[16], intègrent la définition récursive des contraintes. Voir à ce propos PUGET & LECONTE [1995].

❖ Opérateurs Logiques Élaborés

[16] Les processus récursifs correspondent directement aux notions sous-jacentes à la programmation par composants du génie logiciel : à partir de briques, élaborer des maisons qui a leur tour peuvent être utilisées comme briques. Cette méthodologie permet d'éclaircir et de préciser les idées du concepteur du modèle.

Les opérateurs sur les contraintes se sont énormément développés, allant bien au-delà des opérateurs logiques usuels, pour améliorer la conception de contraintes complexes. Un exemple est donné avec l'opérateur de cardinalité :

L'**opérateur de cardinalité** (VAN HENTENRYCK & DEVILLE [1991]), noté #, est un opérateur :

- paramétré par une variable entière et son domaine entier positif fini : $(v, D(v))$, avec $D(v)$ un intervalle de et $\mathrm{card}(D(v)) < +\infty$;
- s'appliquant à un ensemble fini de contraintes C sur un ensemble de variables $V = \{v_1, \ldots, v_n\}$ à domaines quelconques ($\Leftrightarrow C = \{c_1(v_{1,1}, \ldots, v_{1,k(1)}), \ldots, c_m(v_{m,1}, \ldots, v_{m,k(m)})\}$, avec $v_{i,j} \in V$), et
- fournissant une contrainte, notée $\#(v, D(v), \{V, D(V), C\})$, qui est satisfaite par une solution lorsque le nombre de contraintes satisfaites de C est une valeur du domaine de la variable en paramètre :
  Une solution $(d(v), d(v_1), d(v_2), \ldots, d(v_n)) \in D(v) \times D(v_1) \times D(v_2) \times \ldots \times D(v_n)$, satisfait cette contrainte si et seulement si :
  $$d(v) = \mathrm{card}(\{c \in C \text{ tel que : } ((d(v_{c,1}), d(v_{c,2}), \ldots, d(v_{c,k(c)})) \in S(c)\})$$

Un grand nombre de contraintes peuvent être formulées à partir d'un opérateur très évolué – car paramétré par une variable et son domaine – comme l'opérateur de cardinalité.

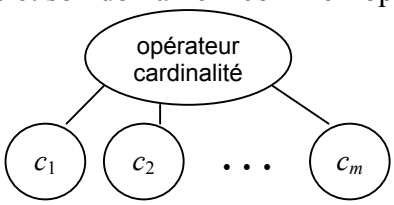


Figure 3 : Représentation graphique des dépendances entre contraintes (récursivité).

Quelle est la nécessité de développer des opérateurs évolués ? Si nous nous limitons aux opérateurs logiques usuels, certaines contraintes complexes nécessitent un nombre exponentiel de ces opérateurs pour être décrites dans toute leur subtilité (la formulation peut être exponentielle en le nombre de contraintes de base, voire en la taille de l'ensemble des données de base). C'est le cas de certaines contraintes obtenues avec l'opérateur de cardinalité.

Les processus récursifs sont très bien adaptés au processus de modélisation utilisé par le concepteur pour établir les modèles de son système, car elle sert de support au paradigme « diviser pour régner » : lorsqu'aucun modèle d'un problème n'est connu du concepteur, alors celui-ci est divisé en problèmes partiels, et ce, de manière itérative, jusqu'à l'obtention de problèmes connus, dits primitifs. Les opérateurs symbolisent les liens identifiés entre les problèmes partiels.

Les opérateurs établissent des liens entre les contraintes pouvant être avantageusement exploités pour déduire des algorithmes de résolution basés sur la suppression de valeurs inconsistantes (voir Annexe A2). Nous en ferons d'ailleurs usage pour la construction de différentes contraintes (voir § 7.4).

Remarquons qu'il n'y a pas unicité du processus récursif : une contrainte peut être l'aboutissement de différents processus récursifs, à partir de différentes contraintes primitives et opérateurs. Inconvénient d'une telle conception des contraintes : l'efficacité de la résolution est fortement dépendant du processus récursif utilisé pour obtenir la contrainte. Pour y remédier partiellement, et pour des combinaisons d'opérateurs très fréquents, il est nécessaire de concevoir des algorithmes spécifiques plus efficaces. Nous en reparlerons au § 7.4.

❖ Contrainte Globale et Problèmes

Une autre caractéristique des contraintes étudiée est la généricité. Certaines contraintes ont un comportement similaire, se traduisant par des sémantiques ne divergeant que par un en-

semble de données pouvant être fournis en paramètres, et des variables entièrement identifiées par ces données.

Ces contraintes peuvent donc être assemblées dans une classe pour former une **contrainte** paramétrée dite **globale**. Chaque valeur de paramètre identifie un élément bien précis de cette classe. On notera $c(\gamma) = (V(c(\gamma)), S(c(\gamma)))$ la contrainte associée à la contrainte globale c ayant comme valeurs de paramètres γ de l'ensemble Γ donné en compréhension.
Comme exemple, nous avons la contrainte induite par l'opérateur de cardinalité. Deux autres contraintes globales sont décrites en Annexe A1 : la contrainte de cardinalité globale GCC et la contrainte globale de différence ALLDIFF. Ces deux contraintes vont être utilisées pour en élaborer de nouvelles aux chapitres 7 et 8. Nous renvoyons à BELDICEANU [2000*a*], ILOG [1995] et COSYTEC [1995] pour différentes listes de contraintes globales.

Étant donnée une contrainte $c(\gamma) = (V(c(\gamma)), S(c(\gamma)))$ avec $\gamma \in \Gamma$, notons $\mathcal{D}_{V(c(\gamma))}$ le produit cartésien des variables de $V(c(\gamma))$. La relation binaire $\Pi(c)$ de $\Gamma \times \bigcup_{\gamma\in\Gamma} \mathcal{D}_{V(c(\gamma))}$ associant à chaque $\gamma \in \Gamma$ un élément de la sémantique $S(c(\gamma))$ peut être vue comme un problème avec Γ l'ensemble des instances et $S(c(\gamma))$ l'ensemble des solutions réalisables de l'instance γ. Par conséquent, une contrainte globale peut être assimilée à un problème.

Cette définition permet de mieux intégrer la complexité des problèmes dans l'étude des contraintes et des systèmes de contraintes. De plus, elle correspond mieux aux modèles utilisés dans la pratique, modélisant des problèmes de l'industrie.

Au chapitre 7, par abus de langage, nous dirons qu'il existe une transformation polynomiale d'une contrainte globale c_1 vers une autre contrainte globale c_2 si toute instance γ_1 de c_1 peut être transformée en une instance γ_2 de c_2 et que $S(c_1(\gamma_1)) = S(c_2(\gamma_2))$.

Plus généralement, la notion de contrainte globale c pourra correspondre à une famille de problèmes dont la définition tourne autour du problème central $\Pi(c)$ (le représentant de la contrainte). Dans cette famille on retrouve tous les problèmes effectuant une « description » des solutions réalisables du représentant $\Pi(c)$. Par exemple :

- la version décision (problème de consistance simple associé) informe si $S(c(\gamma))$ est vide ou non,
- l'optimisation simple informe sur les solutions de $S(c(\gamma))$ qui optimisent une fonction objective,
- le dénombrement informe sur le nombre de solutions réalisables (voir § 2.1.1.2), ou encore
- les différentes variantes de la consistance (voir § 3.2.3.4 et Annexe A2) informent sur les valeurs de domaines qui ne conduisent à aucune solution réalisable.

Cette conception algébrique d'une contrainte a des avantages d'un point de vue complexité (pour une réflexion similaire voir MARATHE [2002]), et permet de concevoir la résolution comme un processus de décision (voir § 3.2.2).

2.1.4.4. Complexité des Systèmes de Contraintes

La plupart des résultats de complexité établis sur les CSP ne concernent que les systèmes de contraintes vérifiant des hypothèses très restrictives.

Le sous-problème le plus étudié est celui dont le système de contraintes $\mathcal{K} = (V, D, C)$ est à domaines finis en extension, et dont l'ensemble $S(c)$ est donné en extension, pour toute contrainte c de C. Dans un tel cadre, le problème de consistance associé (savoir si $S_{\mathrm{CSP}}(\mathcal{K})$ est non vide) est NP-complet (SCHAEFER [1978]). Plusieurs auteurs ont alors caractérisé des sous-problèmes du CSP ayant une complexité polynomiale. Pour une synthèse très détaillée, voir JEAVONS *et al*. [1995], PEARSON & JEAVONS [1997], JEAVONS [1998], BULATOV *et al*. [2000].

Lorsque les contraintes sont données sous forme d'inégalités fonctionnelles ou linéaires, les systèmes de contraintes sont des systèmes d'inégalités en nombres mixtes. Alors les résultats de complexités concernant les MIP sont alors valables.

Dans le cas général, l'utilisation de la définition des contraintes basée sur les problèmes, laisse le champ libre à toutes les possibilités en terme de complexité des problèmes sur les systèmes de contraintes. Il est possible d'établir la complexité du système, connaissant celles des contraintes. Mais certaines précautions doivent être prises. A ce propos, PAPADIMITRIOU [1994 § 14.3] a déjà mis en garde contre les dérives que cela peut entraîner.

Le problème de satisfaction de contraintes recherche une solution réalisable, et de ce fait, ne fait usage d'aucun modèle préférentiel d'évaluation des solutions des systèmes de contraintes. Ce champ de recherche est très récent, et est présenté au prochain paragraphe.

2.1.4.5. Optimisation et Contraintes : Les Contraintes Souples Usuelles

Les systèmes de contraintes sont des modèles de faisabilité (§ 2.1.1.1) et à ce titre ne décrivent que la dichotomie entre les solutions réalisables et les non réalisables. Les modèles proposés en programmation par contraintes, pour supporter une information préférentielle, sont très divers. Ces modèles préférentiels sont formés, entre autre, de contraintes appelées **contraintes souples** (ou **molles** ou **flexibles**) pour les différentier des contraintes du système de contraintes. Les contraintes de faisabilité sont alors appelées par opposition **contraintes dures** (ou **impératives**). Nous présentons ici, les deux contraintes souples principales utilisées dans la littérature : la contrainte souple de propriété et la VC-contrainte souple simple. Divers problèmes d'optimisation faisant usage de ces contraintes souples sont introduits.

❖ La Contrainte Souple de Propriété

Les **contraintes souples de propriété** permettent d'exprimer des préférences bivalentes. Une contrainte souple de propriété peut être représentée par un couple de contraintes (c_p, non[c_p]), où $V(c_p) = V(\text{non}[c_p])$ et $S(\text{non}[c_p]) = \mathcal{D}_{V(cp)} \setminus S(c_p)$. Mais en général, elle est assimilée à la contrainte c_p. Dans ce sens, une contrainte souple de propriété n'est pas satisfaite, si la contrainte c_p ne l'est pas.

La faisabilité d'une solution n'est pas remise en cause selon qu'elle satisfait ou non une contrainte souple de propriété. Les contraintes souples de propriété expriment un but à atteindre : le but est atteint par la solution courante, si elle satisfait la contrainte c_p. Dans la littérature, la contrainte c_p est appelée **but** (LAFAGE & LANG [2001]).

Une contrainte souple de propriété crée une dichotomie entre les solutions : celles qui sont consistantes et celles qui sont inconsistantes avec la contrainte c_p. De même, étant donné un ensemble de contraintes souples de propriété C_p, une solution x crée une dichotomie entre les contraintes souples : celles qui sont satisfaites $C_p(x)$ et celles qui ne le sont pas $C_p \setminus C_p(x)$. Le couple ($C_p(x)$, $C_p \setminus C_p(x)$) est appelé **configuration** (JUSSIEN [1997], JUSSIEN & BOIZUMAULT [1997], BOUTILIER *et al*. [1997], BOUTILIER *et al*. [1999]).

❖ La vc-Contrainte Souple Simple

En programmation par contraintes, une fonction objective f définie sur un ensemble de variables $V(f)$ est représentée par un couple formé de :

– une nouvelle variable v_f appelée **variable valeur** de f, et son domaine de valeurs réelles $D(v_f)$ contenant les différentes valeurs[17] possibles qui peuvent être attribuées aux solutions par la fonction objective f :

[17] L'ensemble des valeurs du vrai-critère possède une structure minimale d'ordre total. Par conséquent, elle peut correspondre à l'ensemble des réels, des entiers, ou à toute échelle qualitative.

$$\{f(x) \text{ tel que } x \in \mathcal{D}_{V(f)}\} = f(\mathcal{D}_{V(f)}) \subseteq D(v_f) \subseteq \quad .$$

– une contrainte c_f, appelée **VC-contrainte souple simple** de f, sur $V(c_f) = V(f) \cup \{v_f\}$ dont l'ensemble réalisable $S(c_f)$ est égal au graphe de la fonction objective f :

$$S(c_f) = \{(x_1, x_2) \in \mathcal{D}_{V(f)} \times D(v_f) \text{ tel que } f(x_1) = x_2\} .$$

Toute contrainte souple de propriété (c_p, non(c_p)) peut être modélisée par une fonction objective équivalente f_p laquelle vaut 1 en la solution $x \in \mathcal{D}_{V(cp)}$ si x est consistante avec c_p, 0 sinon. La construction d'un couple (variable valeur, VC-contrainte souple simple) équivalent est immédiat.

❖ Problèmes et Contraintes Souples Usuelles

Ces contraintes souples ont été combinées pour définir des problèmes d'optimisation. Par exemple le **problème de satisfaction de contraintes maximal** Max-CSP (RÉGIN *et al.* [2001]) admet :

– comme modèle de faisabilité un système de contraintes, et
– comme modèle des préférences un ensemble de contraintes souples de propriété C_p, agrégé au moyen d'une fonction objective cherchant à minimiser le nombre de contraintes souples de propriété non satisfaites.

D'autres contributions ont emprunté des directions différentes pour synthétiser l'information préférentielle véhiculée par un ensemble de contraintes souples de propriété C_p. Ils se ramènent tous à une fonction objective, ou à une ordre partiel sur l'ensemble des configurations. Par exemple, LAFAGE & LANG [2001] considèrent un ensemble de triplets (λ_i, d_i, c_i), avec λ_i un entier naturel, c_i une contrainte souple de propriété de C_p, d_i une distance sur l'ensemble des parties de C_p (ou l'ensemble des configurations). Un exemple de distance est la distance de Hamming d_H, correspondant au cardinal de la différence symétrique entre deux sous-ensembles de contraintes souples de propriété :

Pour tout $C_1, C_2 \in \mathcal{P}(C_p)$, $d_H(C_1, C_2) = \mathrm{card}(C_1 \Delta C_2)$, avec $C_1 \Delta C_2 = (C_1 \setminus C_2) \cup (C_2 \setminus C_1)$

Une fonction objective est ensuite définie en utilisant ces informations.

Pour un approfondissement, nous renvoyons aux références précitées.

D'autres structures sont apparues dans la littérature comme les CSP valués (VCSP, voir SCHIEX *et al.* [1995]), ou les CSP basés sur les demi-anneaux (SCSP = Semiring-Based CSP, voir BISTARELLI *et al.* [1997] et [1999]). Ces modèles ne s'appliquent qu'aux domaines finis en extension. Ils n'utilisent pas de fonctions objectives, mais des moyens d'évaluation plus élaborés, permettant d'élargir les possibilités de modélisation. De par leur jeune âge, ils ont été très peu utilisés dans la pratique. Certaines transformations d'un modèle à un autre ont été étudiées dans BISTARELLI *et al.* [1999]. Notons aussi les travaux très prometteurs de BOUTILIER *et al.* [1997], BOUTILIER *et al.* [1999], sur les préférences conditionnelles.

2.2. Panorama des Problèmes d'Élaboration de Plannings de Travail de Personnel

Les trois types de modèles théoriques précédents forment la base de l'essentiel des modèles, traitant de planification d'horaires de travail, accessibles dans la littérature. Ces modèles ont conduit, pour la plupart, à une réalisation logicielle apportant des réponses informatiques aux problèmes de planification d'horaires de travail. Par exemple Interdip (ABDENNADHER & SCHLENKER [1999]), Equitime (CHAN [2002]), Gymnaste (WEIL *et al.* [1994], HEUS [1996], CHAN *et al.* [1998], CHAN [2002]), etc. Le concepteur de logiciels d'optimisation ILOG, pro-

pose à partir de son site internet[18], une série d'articles présentant des logiciels de planification d'horaires de travail conçus à partir de ses bibliothèques basées sur les CSP (`IlogSolver`) et les MILP (Cplex).

Les problèmes d'élaboration de plannings de travail de personnel sont très variés. Il est nécessaire de les catégoriser pour mieux les comprendre et leur apporter des solutions. Ils sont très souvent classés selon l'activité de l'unité cible. Par exemple le milieu hospitalier, le personnel aéroportuaire, les conducteurs de train, etc. Cette classification est toujours très utilisée, mais n'est pas suffisamment significative pour l'automatisation.

D'autres classifications reposent sur différentes entités pertinentes des problèmes. Par exemple : certaines caractéristiques des tâches à effectuer, les caractéristiques organisationnelles, l'horizon de planification, l'hétérogénéité du personnel, etc.

Cette seconde partie de chapitre se déroule comme suit : Après avoir présenté les objets de base d'un problème d'élaboration de plannings de travail de personnel (§ 2.2.1), nous détaillons (§ 2.2.2) la classification utilisée par PARTOUCHE [1998]. Un état de l'art (du § 2.2.3 au § 2.2.8) est ensuite effectué à partir d'une classification plus appropriée aux différents modèles utilisés. Notons que cet état de l'art ne fait que compléter la liste des articles classés par PARTOUCHE [1998], son état de l'art étant récent et très riche (plus de 120 références).

2.2.1. Concepts et Notation

Pour établir un modèle du problème d'élaboration de plannings de travail de son personnel, l'unité autonome se base sur un certain nombre d'objets dont les principaux sont présentés ci-dessous. Les objets plus secondaires sont introduits à mesure de leur utilisation. Un grand nombre de définitions s'inspirent de la thèse de PARTOUCHE [1998, Annexe A].

2.2.1.1. Les Éléments (ou Objets) de Base

Une **période** est un plus petit intervalle de temps non vide pour lequel la charge est définie. Toute période est notée p, et l'ensemble des périodes P. Chaque période p a une date de début $\text{début}(p)$, une date de fin $\text{fin}(p)$ et une durée $\text{amplitude}(p)$, qui, selon le problème, constituent des informations pertinentes, connues ou non. La durée d'une période est plus ou moins petite et dépend du niveau de précision souhaitée et possible pour la planification. Par exemple, le planning journalier des caissiers admet des périodes allant de 10 minutes à une heure. Le secteur bancaire par contre fait usage de périodes correspondant à une journée de travail. La période indique le niveau de détails recherché pour le planning, et à ce titre peut s'identifier à l'intervalle de temps d'une tâche précise ou à celle d'un ensemble de tâches. Les caractéristiques communes aux tâches associées à une période caractérisent celle-ci. Remarquons que la notion de période ici ne correspond pas à celle d'intervalle de temps car un planning regroupe plusieurs périodes qui peuvent se chevaucher dans le temps.

Une **tâche** t est une entité de base à couvrir par le travail des salariés. Elle est caractérisée par un ensemble d'attributs spécifiques, connus ou non, et pertinents pour le système étudié. Par exemple :

- une heure de début $\text{début}(t)$ et une heure de fin $\text{fin}(t)$;
- une localisation de début et de fin (cas du personnel mobile) ;
- le site de réalisation (cas multi-site) ou/et l'unité concernée (cas multi-unité) ;
- les ressources (de la tâche) nécessaires à son bon déroulement : qualifications, équipes ;

[18] http://www.ilog.com

– les préférences et dépendances entre les différentes équipes ou qualifications pouvant effectuer cette tâche ; etc.

L'ensemble des tâches est notée *T*. Lorsque des tâches courtes ou sécables nécessitent une même équipe mono-qualifiée (ou qu'elles s'y ramènent), elles sont traduites en une courbe de charge ; Lorsqu'elles comportent des imprécisions, elles sont traduites en courbe de charge et table de besoins ; Et dans le cas contraire, les tâches sont utilisées directement. La période $p(t)$ sous-jacente à une tâche correspond à l'intervalle de temps [début(t), fin(t)].

Une **pause** (*Break*) est un intervalle de temps non travaillé en cours de journée, correspondant à une coupure, une pause-repas ou une pause-café.

Un **repos** (*Day-off*) est un jour non travaillé. Bien qu'indissociables au niveau horaire, les repos sont classés selon leur fonction dans les plannings. Par exemple, repos compensateur, repos hebdomadaire, congé annuel, congé maladie, congé RTT (en France, pour certaines catégories de personnel, les 35 heures sont traduites par des jours de repos supplémentaires), etc. Certains repos (repos compensateurs, RTT en France) sont répertoriés comme des périodes de 8 heures de travail effectif. Par exemple, une semaine de 40 heures de travail effectif peut se décomposer en un repos compensateur et 4 jours de 8 heures en entreprise.

Un **horaire de travail** sur un horizon de planification donné, spécifie les périodes de travail – et par conséquent de non travail (repos, pauses) – d'un individu. Un horaire est noté *h*, l'ensemble des horaires *H*. Ces entités possèdent un ensemble d'attributs caractéristiques que sont une date de début début(h) et de fin fin(h), une amplitude (amplitude(h) = début(h) – fin(h)) et une durée travaillée durée travaillée(h). La définition de la durée travaillée d'un horaire dépend du problème traité. Elle correspond dans certains cas, à la somme des durées des périodes effectives de travail. Dans d'autres cas elle dépend de la localisation de la période dans la journée (heures d'équivalence), ou du jour de semaine (le travail le week-end peut donner droit à des repos compensateurs). Dans d'autres encore, elle correspond à la durée du travail effectif (incluant par exemple les repos compensateurs, etc.)

Quelques horaires de travail particuliers :

– **Vacation** (*Shift*) : horaires de travail limités à la journée, précisant les pauses[19].
– **Tour** (*Tour*) : enchaînement de vacations et de repos sur un horizon dont la durée est de l'ordre de la semaine.

Un **programme individuel de travail** spécifie l'ensemble des tâches à réaliser par un individu sur un horizon de planification donné. L'ensemble des périodes sous-jacentes aux tâches définit un horaire de travail.

Une **étiquette** est un élément d'une partition arbitraire de l'ensemble des vacations et repos possibles sur une journée. Une étiquette est notée *ev* et l'ensemble associé *EV*. Généralement, les différents repos forment à eux seuls une étiquette, contrairement aux vacations. Chaque élément d'une partition de l'ensemble des vacations est appelé **type de vacation**. Un exemple de division des vacations : Matin, Soir, Nuit. Ces sous-ensembles de vacations sont définis à partir d'attributs, par exemple l'amplitude, l'heure de début, etc.

Une **équipe** est une liste de couples (nombre d'employés, qualification) exprimant une charge nécessitant quelques fois une coordination entre plusieurs tâches.

[19] Selon le système étudié, la vacation est appelée service (chauffeurs de bus, ou personnel naviguant des compagnies ferroviaires et aériennes), ou code horaire, ou tranche horaire (personnel hospitalier).

La **charge** (*Requirements*, *Demand*) est l'ensemble des informations du problème exprimant les besoins à satisfaire au mieux. Elle traduit une forme de compromis entre la qualité de service et les coûts, et prend la forme de :

- Une **courbe de charge** (Figure 4), lorsqu'elle s'exprime sur un horizon d'une journée par un nombre nécessaire (suffisant ou idéal) d'employés par période de durée généralement de 10 minutes à une heure[20]. En d'autres termes, pour tout $p \in P$, nous connaissons le besoin nécessaire ($d_{\min}(p)$), suffisant ($d_{\max}(p)$) ou idéal ($d(p)$) ;
- Une **table de besoins** lorsqu'elle s'exprime par un nombre nécessaire (suffisant ou idéal) d'employés par jour (ou par couple étiquette-jour, etc.)[21] ;

Étiquettes	lundi	mardi	mercredi	jeudi	vendredi	samedi	dimanche
Matin	5	5	4	5	6	5	7
Soir	1	1	2	2	2	3	1
Nuit	1	1	1	1	1	1	1

Tableau 2 : Exemple d'une table de besoins (d'après PARTOUCHE [1998, tab. 6.1])

- Un ensemble d'équipes admissibles par tâche, et dans ce cadre :
  - l'ensemble des contraintes liant les équipes d'une tâche à une autre (**contraintes entre équipes**) ;
  - l'ensemble des dépendances (hiérarchie) entre les différentes équipes aptes à effectuer une tâche[22] (**préférences entre équipes**) ;
- Des préférences traduisant un compromis entre la qualité de service et les coûts.

La charge peut être cyclique, c'est-à-dire qu'elle suit une certaine périodicité, souvent hebdomadaire. La grande majorité des travaux utilisant les tables de besoins considèrent une charge cyclique.

[20] Ces courbes sont généralement obtenues à partir de simulations basées sur la théorie des files d'attente, avec notamment les courbes d'Erlang (SAATY [1961], COTÉ & STEIN [2000], SEGAL [1974]).

[21] Ces tables sont obtenues de différentes manières. Par exemple en planification infirmière, la table de besoin est le résultat de l'agrégation de plus d'une trentaines d'indicateurs portant sur les patients. Ce problème est connu en RO sous le nom de *staffing problem* (WARNER [1976]), et les systèmes informatiques existants portent le nom de *workload management systems* (SIFERD & BENTON [1992]).

[22] Ces préférences traduisent par exemple la possibilité de remplacer un employé d'une compétence par un autre employé d'une autre compétence, avec une préférence pour le premier.

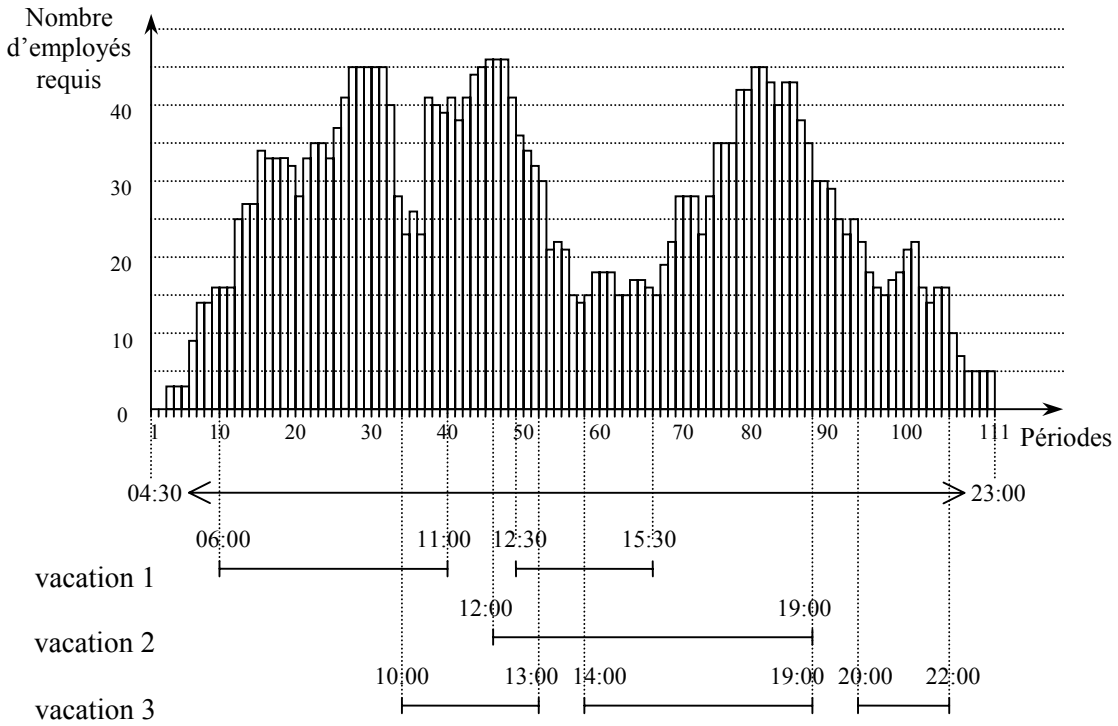


Figure 4 : Exemple d'une courbe de charge d'un jour des Aéroports de Paris (ADP) sur 111 périodes, et de trois vacations réalisables (PARTOUCHE [1998]).

Une **classe de personnel** (ou groupe d'employés) est un ensemble d'employés ayant même contrat de travail (en excluant les alinéas non pertinents pour les plannings : clauses de non concurrence, exclusivité, identité du salarié, etc.) et soumis aux mêmes règles d'usage. Les propriétés pertinentes sur lesquelles repose la partition des employés en classe sont appelées **ressources de la population** (ou attributs pertinents pour cette population). Exemples d'attributs pertinents

- durée de travail (temps complet ou partiel à 95%, 90%, …),
- niveau de qualification,
- travail temporaire, apprentissage, CDI, CDD, étudiants, permanents
- site ou unité de référence (planification multi-site, personnel mobile avec découché, et multi-unité),
- horaires individualisés (travailleurs de nuit, de week-end, etc.), politique de rémunération des heures supplémentaires, ancienneté, etc.

Notation : l'ensemble des classes de personnel est noté $\mathcal{G}$, et l'ensemble des propriétés associées est noté entre crochets. Par exemple, une classe de personnel se distinguant par des durées légales de travail hebdomadaire D_{LTH} et de compétences S est noté $\langle D_{LTH}, S\rangle$. L'identification du groupe ayant une durée légale et une compétence données se fait par : $G^{\langle \delta, s\rangle}$ avec $\delta \in D_{LTH}$ et $s \in S$. Celle du groupe des personnes dont les contrats de travail spécifient la même compétence $s \in S$, est notée $G^{\langle s\rangle}$. Et un groupe quelconque de $\mathcal{G}$ est noté G.

2.2.1.2. Contraintes de Définition des Horaires de Travail

Les plannings de travail sont généralement soumis à un ensemble de contraintes caractérisant les horaires de travail admissibles. Ces contraintes proviennent des règles de droit régissant chaque profession, ainsi que des us et coutumes de l'entreprise. Certaines contraintes reviennent fréquemment et ont été classées en trois groupes par PARTOUCHE [1998]. Nous y

ajoutons deux autres groupes formant les contraintes de cyclicité des horaires et les contraintes spécifiques aux tâches :

Les **contraintes de positionnement des repos** définissent les séquences de jour de repos et de travail qu'il est possible de trouver dans un planning. Elles imposent des bornes sur le nombre de repos pour différentes périodes de jours (la semaine, le mois, le cycle), des bornes sur le nombre de jours de repos consécutifs, des bornes sur la proportion de week-ends de repos, des bornes sur le nombre de jours de travail successifs, des bornes sur la durée de travail en fonction de différentes périodes de jours (semaine, mois, cycle, horizon), etc.

Les **contraintes de positionnement des étiquettes** imposent des bornes sur la durée entre deux vacations successives, des bornes sur le nombre de jours successifs de travail selon une étiquette donnée (pas plus de 4 horaires de nuit successifs), inclus des contraintes de changement de vacations définissant pour chaque étiquette une liste d'étiquettes pouvant lui succéder, etc.

Les **contraintes de définition des vacations admissibles** imposent des bornes sur l'amplitude, des bornes sur la durée travaillée, des bornes sur l'heure de début et de fin d'une vacation, un nombre de pauses et leur positionnement, des bornes sur la durée des pauses, des bornes sur les durées de travail en continu, etc.
La Figure 5 montre un exemple de paramètres pouvant entrer dans la définition des contraintes d'une vacation admissible.

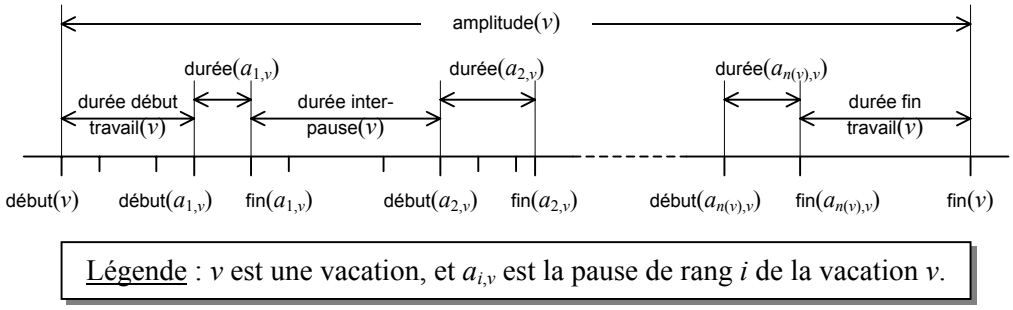


Figure 5 : Différents paramètres d'une vacation *v*.

Les **contraintes de cyclicité des horaires de travail** imposent une périodicité des horaires individuels, une équivalence des horaires d'un employé avec ceux d'un autre s'exprimant par un décalage d'un horaire par rapport à l'autre ; etc.
Quelques remarques :

1) Les contraintes de cyclicité permettent de construire des plannings dont l'horizon de planification n'est pas fini ; la périodicité des horaires qui induisant celle des plannings.
2) L'équivalence des horaires de travail à un déphasage près n'implique pas une évaluation égale de ces horaires, bien que cette hypothèse soit souvent admise dans la littérature.

Les **contraintes de tâches** regroupent les contraintes de précédence, les contraintes de transition minimales entre deux tâches, les contraintes de disjonction et date au plus tôt ou au plus tard lorsque la planification des tâches effectuée en amont, laisse des degrés de liberté sur la date effective de début de certaines tâches, pour améliorer le planning de travail des employés, ou encore lors de la gestion simultanée des réunions et formations (HERTZ *et al.* [2000]).

Les horaires de travail dépendent aussi d'autres facteurs, comme les attributs des employés et les ressources supplémentaires nécessaires à la réalisation des tâches. Ces concepts généraux servent à la définition de problèmes d'élaboration de plannings de travail de personnel.

2.2.2. *Typologie des Problèmes d'Élaboration de Plannings de Travail*

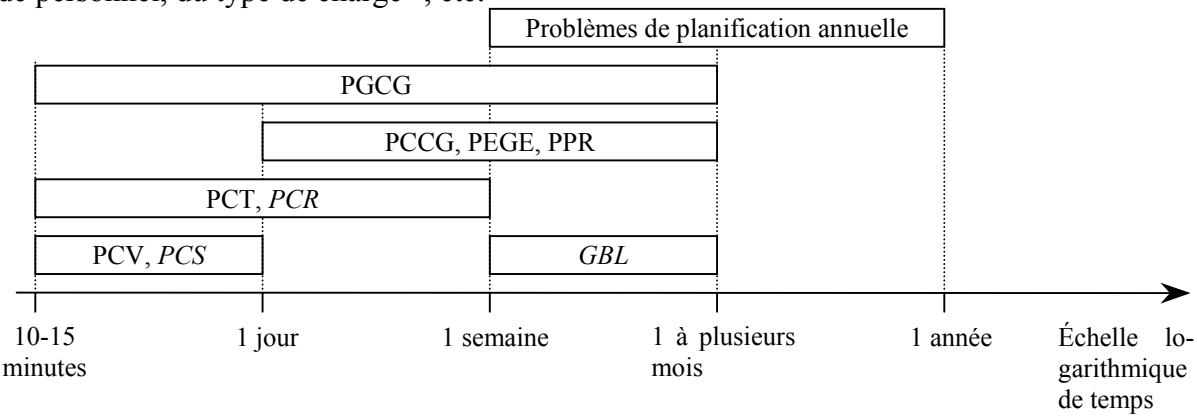
Tous ces éléments de base mettent en évidence la variété des PEPTP pouvant exister. Selon l'importance accordée aux éléments de base, nous pouvons créer différentes classifications de ces problèmes. PARTOUCHE [1998] propose une classification donnant la priorité à l'horizon de planification et au niveau de détail initial. Dans ce document, nous l'appelons classification inspirée des activités. Elle est présentée à la prochaine section. Une autre classification, mettant en avant la connaissance (explicite ou non) du nombre d'employés et des horaires de travail, est ensuite présentée à la section 2.2.2.2. Elle est dite inspirée des modèles.

2.2.2.1. *Classification Inspirée des Activités (PARTOUCHE [1998])*

Cette classification met en avant trois attributs qui sont :

- la durée (en moyenne ou minimale) des périodes (10 minutes à une semaine),
- la durée de l'horizon de planification (un jour à un an),
- les types de tâches formant les périodes (sécables, non sécables).

Puis, en seconde priorité, elle tient compte des contraintes horaires, des contraintes de classe de personnel, du type de charge[23], etc.



Légende :

- Les problèmes identifiant les mêmes attributs horaires sont représentés par un même rectangle. Le coté gauche désigne la durée approximative des périodes du planning, et le coté droit désigne la durée approximative de l'horizon.
- Les sigles de problèmes sont définis dans le corps du texte.
- Les sigles en italiques identifient les problèmes dont les périodes sont des tâches.

Figure 6 : Classification inspirée par les activités, de quelques PEPTP (Déduite de la figure 1.3 de PARTOUCHE [1998]).

PARTOUCHE identifie plusieurs sous-problèmes très étudiés dans la littérature. Ces problèmes intègrent entre autre des contraintes de charge et des informations sur la ou les classes de personnels envisagées. Nous les décrivons ci-dessous :

Le **problème de construction de vacations** (PCV, *shift scheduling problem*) consiste à déterminer un ensemble de vacations réalisables qui couvre le mieux possible une courbe de charge, tout en minimisant les coûts.

Le **problème de construction de tours** (PCT, *tour scheduling problem*) consiste à déterminer un ensemble de tours admissibles qui couvre le mieux possible une courbe de charge (défini généralement sur une semaine), tout en minimisant les coûts. Les horaires réalisa-

[23] La charge peut être exprimée en contraintes ou/et préférences. De plus, plusieurs sous-problèmes peuvent être étudiés en limitant par exemple la forme des courbes de charges considérées, etc.

bles (ou tours dans ce cadre) satisfont des contraintes de définition de vacations admissibles, de jours de repos et d'enchaînement des vacations.

Le **problème de positionnement des repos** (PPR, *days-off scheduling problem* ou *single-shift scheduling problem*) recherche un planning de repos soumis aux contraintes suivantes : (*a*) La charge est décrite par un besoin en personnel par jour (table de besoins) : $d(j)$ pour tout jour j de l'horizon ; (*b*) Elle admet des contraintes de positionnement des repos.

Le **problème d'élaboration de grilles étiquetées** (PEGE, *multiple shift scheduling problem*) consiste à déterminer un planning de travail soumis aux contraintes suivantes : (*a*) Une charge à couvrir, décrite par un besoin en personnel par étiquette et par jour ; (*b*) Des contraintes de positionnement des repos et de positionnement d'étiquettes, (*c*) Des contraintes de cycle dans le cas de plannings cycliques, etc.

Le **problème central de construction de grilles** (PCCG) se propose de déterminer un planning (*a*) en considérant une charge décrite par une table de besoins en personnel par étiquette (ou vacation) et par jour (table de besoins), (*b*) en admettant des contraintes de positionnement des repos et de positionnement des étiquettes, (*c*) le personnel étant homogène.

Le **problème global de construction de grilles** (PGCG) détermine un planning à partir (*a*) d'une charge décrite par une courbe de charge par jour formant l'horizon de planification, (*b*) des contraintes de définition des vacations admissibles, de positionnement des repos et de positionnement des étiquettes, et (*c*) où plusieurs classes de personnel (temps plein, temps partiel) sont admises pour couvrir la charge.

Le **problème de construction de services** (PCS, *crew scheduling problem*) consiste à déterminer une liste d'horaires réalisables qui couvre au moindre coût un ensemble de tâches de durée inférieure à la journée (ex. : tronçons de lignes de transports urbains, tronçons de vol pour les court-courriers).

Le **problème de construction de rotations** (PCR, *crew pairing problem*) consiste à déterminer une liste d'horaires réalisables (rotations) qui couvre au moindre coût un ensemble de tâches de durée égale à la journée (ex. : tronçons de vol pour les long-courriers).

Le **problème de génération de Bid-Lines** (GBL, *bid-line generation*) consiste à déterminer une liste d'horaires réalisables (*bid-lines*) qui couvre au moindre coût un ensemble de tâches de durée 3 à 7 jours (rotations) sur un horizon mensuel.

Cette classification est d'abord historique et souffre des limites inter-disciplinaires. Par exemple, les problèmes d'élaboration de plannings infirmiers (*nurse scheduling problem*) sont des PEGE généralisés (la charge pouvant être exprimée autrement que par une contrainte (voir par exemple § 2.2.3.4)), pourtant ce type de problème peut être appliqué aux employés de banque, au personnel pénitencier, etc.. De plus, la durée des périodes et de l'horizon jouent un rôle assez négligeable dans la définition des modèles, et ne devrait pas être un critère primordial de classification. Notons enfin qu'avec l'apparition de problèmes « multi-niveau » de planification, comme les PGCG intégrant une planification à la fois journalière et mensuelle (PARTOUCHE [1998], CHAN [2002]), l'accent devra être mis en priorité sur les contraintes de même sémantique (§ 2.1.4.3) situées à des niveaux de planification différents, et les contraintes inter-niveau. Les modèles multi-niveau intègrent aussi un autre aspect très important qui est la précision des données et des résultats, aspect pas du tout abordé dans la littérature, mais qui à notre avis est essentiel pour effectuer une aide à la décision efficace.

2.2.2.2. Classification Inspirée des Modèles

De manière à mieux refléter les différents modèles dans la classification, la priorité a été donnée aux deux attributs suivants :

– la connaissance de la taille de l'ensemble des employés[24],
– la donnée de l'ensemble des horaires individuels en extension ou en compréhension.

Ces deux attributs sont ensuite complétés par les autres concepts, comme les classes de personnel, les contraintes de définition des horaires de travail lorsque le second attribut est en compréhension, et les critères d'évaluation : la charge, les coûts, les souhaits individuels, etc.

Cette classification répartit en quatre classes les problèmes d'élaboration de plannings. Elle est résumée[25] dans le Tableau 3.

Les **problèmes d'élaboration de plannings de type affectation** (PEPTA, *Assignment-Type Timetabling Problems*) supposent connus le nombre d'employés et l'ensemble des horaires individuels, et cherchent par conséquent à sélectionner un horaire admissible pour chaque employé, définissant un planning établissant un meilleur compromis entre les préférences des différents acteurs ;

Les **problèmes d'élaboration de plannings de type affectation généralisée** (PEPTAG, *Generalized Assignment-Type Timetabling Problems*) supposent connu le nombre d'employés et cherchent à construire leurs horaires de travail (ou programmes individuels de travail), de manière à obtenir un planning établissant un compromis de qualité entre les préférences des différents protagonistes ;

Les **problèmes d'élaboration de plannings de type recouvrement d'ensemble généralisé** (PEPTREG, *Generalized Set Covering-Type Timetabling Problems*) supposent connu l'ensemble des horaires individuels et cherchent à dimensionner l'effectif et à sélectionner les horaires de travail qui seront affectés à chacun, pour identifier un planning établissant un compromis de qualité entre les préférences des différents protagonistes ;

Les **problèmes d'élaboration de plannings de type implicite** (PEPTI, *Implicit-Type Timetabling Problems*) cherchent aussi bien à déterminer l'effectif nécessaire à la réalisation satisfaisante des tâches à effectuer, et à construire les horaires de travail (ou programmes individuels de travail). Le but final est d'identifier un planning établissant un meilleur compromis entre les préférences des différents acteurs.

Les PEPTREG et PEPTI sont souvent qualifiés d'impersonnels, car le nombre total d'employés étant inconnu, pour ne pas privilégier le sous-ensemble d'employés qui serait connu, les souhaits individuels ne sont pas admis. Deux employés sont différenciés uniquement par rapport à leur classe de personnel. Les PEPTA et PEPTREG n'ont pas de contraintes de définition des horaires de travail admissibles, ceux-ci étant donnés en extension.

		Nombre d'employés	
		connu	inconnu
Horaires individuels sur l'horizon	connus	PEPTA	PEPTREG
	inconnus	PEPTAG	PEPTI

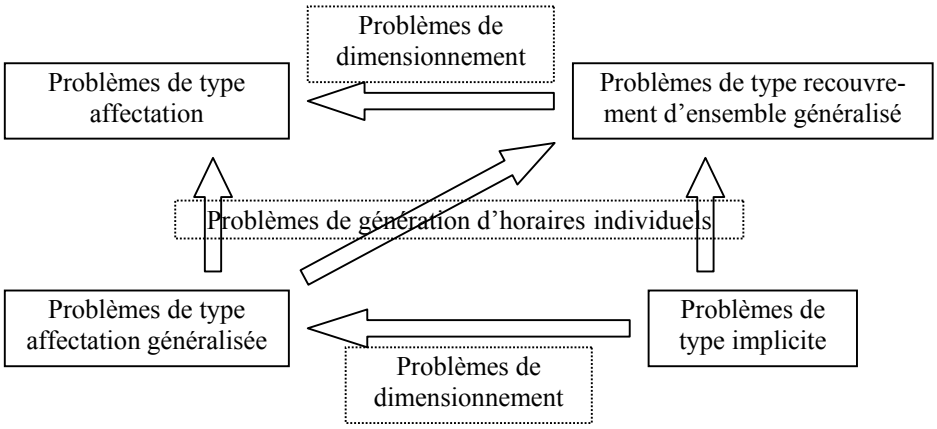
Tableau 3 : Classification des PEPTP en fonction de la connaissance explicite des horaires individuels et du nombre d'employés.

Les sections qui suivent (§ 2.2.3 à 2.2.6) détaillent ces quatre types de problèmes et donnent des exemples de problèmes de la littérature.

[24] Remarquons que dans la pratique, il est équivalent de dire que card(E) est connu ou que E est connu en extension, bien qu'en termes de codage la première assertion occupe une place en O(log(card(E))), alors que la seconde occupe une place en O(card(E)). Mais généralement les contraintes de capacité et de charge sont de l'ordre de O(card(E)). Nous y reviendront au § 2.2.8.

[25] PARTOUCHE [1998] a implicitement identifié ces quatre classes.

Un certain nombre de transformations ont été effectuées dans la littérature pour passer d'un PEPTP à un autre. Ces transformations se classent en deux catégories formant les **problèmes de dimensionnement** des effectifs et les **problèmes de génération d'horaires de travail** (ou programmes individuels de travail). La Figure 7 illustre ces relations entre type de problèmes. Ces deux types de problèmes sont détaillés aux § 2.2.7 et 2.2.8.



Légende :
- les cadres en traits continus identifient les problèmes d'élaboration de plannings de personnels.
- les pointillés encadrent les problèmes permettant de passer d'un problème d'élaboration à un autre.

Figure 7 : Relations entre les types de problèmes d'élaboration de plannings de travail de personnel, induite par la classification inspirée des modèles.

Cette classification est très pertinente, et met en avant une classification se couplant mieux aux différents modèles utilisés pour représenter les problèmes d'élaboration, et ce, quel que soit l'horizon et les périodes utilisés. Bien entendu, ces deux classifications sont complémentaires. Mais étant concerné d'abord par la classification des modèles informatiques utilisés, et non par l'activité de l'entreprise, l'état de l'art est effectué avec la classification inspirée des modèles.

Avant de continuer, nous utiliserons une convention pour identifier les problèmes d'élaboration de plannings très spécifiques présentés par différents auteurs. Le type de problème est spécifié suivi des initiales du nom des différents auteurs entre parenthèse. Par exemple, le problème d'élaboration de planning de type affectation présenté par WARNER dans son article [1976] est noté PEPTA(W).

2.2.3. Problèmes d'Élaboration de Planning de Type Affectation (PEPTA)

Ce type de modèle est utilisable lorsque l'ensemble des employés est connu, ainsi que l'ensemble des horaires (ou programmes individuels de travail). Le but est de sélectionner un horaire par employé pour former un planning effectuant un meilleur compromis entre les préférences des différents acteurs. Après avoir introduit les éléments caractéristiques des PEPTA, nous présentons en détail trois problèmes différents souvent cités dans la littérature.

2.2.3.1. Éléments Caractéristiques

Dans sa forme la plus simple, un problème d'élaboration de planning de type affectation suppose connus :

E	ensemble fini de travailleurs en extension
P	ensemble fini de périodes en extension
$H(e)$	ensemble fini d'horaires en extension pour chaque employé e de E

$a(e, h, p)$ booléen égal à 1 si p est une période de travail pour l'horaire h de l'employé e, et 0 sinon. Donne une description de chaque horaire h par ses périodes de travail

Le 4-uple $(E, P, \{H(e) : \forall\ e \in E\}, \{a(e, h, p) : \forall\ (e, p) \in E \times P \text{ et } \forall\ h \in H(e)\}) = I_{\text{PEPTA}}$ forme les informations minimales de ce type de modèle.

❖ Modèles Basés sur les Systèmes d'Inégalités en Nombres Mixtes

Avec les modèles basés sur les systèmes d'inégalités en nombres entiers, un planning x est décrit par les variables bivalentes $x(e, h)$ valant 1 pour indiquer que l'horaire h de $H(e)$ est affecté à l'employé e dans x, et 0 sinon. L'ensemble des solutions réalisables $S(I_\Pi)$ d'un problème Π entrant dans ce cadre vérifie la contrainte d'affectation :

$$S(I_\Pi) \subseteq S(I_{\text{PEPTA}}) = \left\{ x \in \quad {}^{ns} \left| \begin{array}{l} \sum_{h \in H(e)} x(e,h) = 1 \quad \text{pour tout } e \in E \\ x(e,h) \in \{0,1\} \quad \text{pour tout } h \in H(e) \quad \text{et} \quad e \in E \end{array} \right. \right\},$$

avec $ns = \sum_{e \in E} \text{card}(H(e))$.

En d'autres termes, dans un planning, un employé n'est affecté qu'à un seul horaire.
Les informations P et $a(e, h, p)$ sont utilisées pour définir la qualité de service et le coût global engendré par un planning[26].

❖ Modèles Basés sur les Systèmes de Contraintes

Avec les modèles basés sur les systèmes de contraintes, deux autres solutions sont possibles pour décrire un planning x :

- utiliser des variables $x(e)$, pour tout employé e, à valeur dans $H(e)$, ou
- utiliser des variables $x(h)$, pour tout horaire h de $\bigcup_{e \in E} H(e)$, à valeur dans $\mathcal{P}(E(h))$, avec $E(h)$ l'ensemble des employés pouvant être affecté à h.

D'un point de vue pratique, la seconde représentation définit un espace de recherche sensiblement moins volumineux que la première[27], lorsque le nombre d'horaires est supérieur au nombre de sous-ensembles d'employés. Mais lorsqu'aucune précision n'est donnée, dans le cas général la seconde représentation n'est pas polynomiale en la taille des données. Remarquons que la définition des variables exprime implicitement la contrainte d'affectation d'un seul horaire à un employé.

❖ Évaluation des Plannings

En plus du quadruplet d'ensembles caractérisant les PEPTA, d'autres informations sont considérées comme les classes de personnel, les unités ou encore les étiquettes, etc. Les informations complémentaires ne contribuent pas seulement à la définition des périodes (comme les étiquettes, etc.), elles servent également à formuler :

- des contraintes de capacité par exemple (le nombre de postes de travail est limité), et
- des critères : la qualité de service, la satisfaction sociale de la population et le coût induit par chaque planning.

Nous allons maintenant présenter quatre exemples de modèles : le modèle standard de FERLAND & LAVOIE [1992], le modèle de JAUMARD *et al.* [1998], puis celui de MILLER *et al.* [1976] suivit de WARNER [1976].

---

[26] A titre d'information, les PEPTA sont des **problèmes de type affectation** (ATP : *Assignment-Type Problem*). Voir à ce propos FERLAND *et al.* [1996] et FERLAND & MARZILIANO [2001].

[27] L'estimation de la taille de cet espace est donnée par le produit des tailles de domaines (voir formule (2–1))

2.2.3.2. Le Modèle Standard de Ferland & Lavoie [1992]

Le problème PEPTA(FL) de FERLAND & LAVOIE [1992] considère les données supplémentaires suivantes :

$ct(e, h)$ coût induit par l'affectation du travailleur e à l'horaire h, pour chaque h de $H(e)$ et pour chaque employé e de E,

$d_{\max}(p)$ demande maximale en employés pour chaque période p.

Le problème PEPTA(FL) est un MILP qui identifie le coût d'un planning à la somme des coûts horaires et traduit la qualité de service en contraintes de borne supérieure en chaque période (pas de sureffectif) :

PEPTA(FL) : Étant donné un triplet (I_{PEPTA}, $\{ct(e, h) : \forall\ e \in E$ et $\forall\ h \in H(e)\}$, $\{d_{\max}(p) : \forall\ e \in E\}$) dont les éléments sont définis comme précédemment, trouver une solution x vérifiant : $x \in S(I_{\text{PEPTA}})$ et $\sum_{e \in E} \sum_{h \in H(e)} a(e,h,p) \cdot x(e,h) \leq d_{\max}(p)$ pour tout $p \in P$,

et qui minimise la fonction $\sum_{e \in E} \sum_{h \in H(e)} ct(e,h) \cdot x(e,h)$, si le domaine réalisable est non vide. Sinon retourner "non".

Des variantes sont obtenues en traduisant la qualité de service par des bornes inférieures ou par des bornes de sureffectif et de sous-effectif. Prenons l'exemple de JAUMARD *et al.* [1998] :

2.2.3.3. Le Problème Maître de JAUMARD et al. [1998]

Le problème maître de JAUMARD *et al.* [1998] considère plusieurs classes de personnel caractérisées par un type de contrat (infirmiers permanents à temps plein, infirmiers permanents à temps partiel, infirmiers intérimaires) et une compétence $G^{\langle \delta, s \rangle}$. De plus, ce problème considère un ensemble de jours J de périodes de jour PJ et de type de vacation TV. Notons $p = (pj, j)$ la période de jour pj du jour j. La description d'un horaire h est effectuée par le booléen $a(e, h, p, s, tv)$ à la place du booléen $a(e, h, p)$ du modèle standard. $a(e, h, p, s, tv)$ vaut 1 ssi pour l'horaire h, l'employé e de niveau de qualifications s travaille le type de vacation tv contenant la période p. Pour exprimer la charge, ce problème utilise la notion d'équipe. Une équipe est modélisée par un couple $(S_1, TV_1) = eq$ de sous-ensembles respectivement de S et TV. Pour une période $p \in PJ \times J$ donnée, une demande $d(eq)$ est formulée pour un ensemble $EQ(p)$ d'équipes. Selon le cas, cette demande admet ou non un sureffectif et/ou sous-effectif.
La fonction de coût tient compte à la fois du salaire de chaque employé $ct(e, h)$, de leurs préférences $q(e, h)$ pour chaque horaire (quota de points attribué par e à h), et des préférences $q(s, p)$ entre les différentes qualifications pour une période donnée (valeur donnée au fait d'affecter un employé de compétences s à la période p).

Ce type de modèle représente la majorité des représentations utilisées dans la littérature, mais d'autres solutions ont été proposées. Voici maintenant deux modèles qui font preuve d'originalité dans la représentation des données adoptée.

2.2.3.4. Élaboration de Plannings Infirmiers selon MILLER et al. [1976]

Le problème de MILLER, PIERSKALLA et RATH PEPTA(MPR) est dédié à l'élaboration de plannings infirmiers. L'originalité de PEPTA(MPR) par rapport à PEPTA(FL) est la prise en compte d'une satisfaction sociale, et de la traduction des trois critères d'évaluation des plannings en fonctions objectives. Les employés sont impliqués dans la conception du modèle. L'établissement de la liste des horaires $H(e)$ pour chaque employé e suit une procédure non programmée nécessitant une participation active de e. La période étant le jour ($\Leftrightarrow P = J$), un horaire est défini en accordant la priorité aux requêtes spéciales (disponibilité) de l'infirmier. Les contraintes de positionnement des repos (conventions collectives) et l'assurance au ni-

veau de la continuité entre les horaires admissibles $H(e)$ et des horaires précédemment planifiés et à venir, sont ensuite prises en compte dans la mesure du possible.

Pour élaborer un modèle préférentiel, ce problème considère les informations complémentaires suivantes :

$\mathcal{G}$ ensemble de classes de personnel effectuant une partition de E ;

$(d_{\min}(p), d(p), f_p)$ triplet de pénalité de charge pour chaque période p valable pour l'ensemble des employés. Ce triplet est définit comme suit :

$d_{\min}(p)$ est le nombre minimum de personnes à planifier pour la période p ; $d(p)$ est la valeur idéale pour la charge de la période p ; et f_p est une fonction convexe, linéaire par morceaux (Figure 8) exprimant le coût engendré par la charge lorsqu'elle s'écarte de sa valeur idéale. Le minimum pour f_p est par conséquent atteint en $d(p)$. f_p est appelée fonction de coût de pénalité de charge exprimant à la fois une qualité de service et un coût par période p.

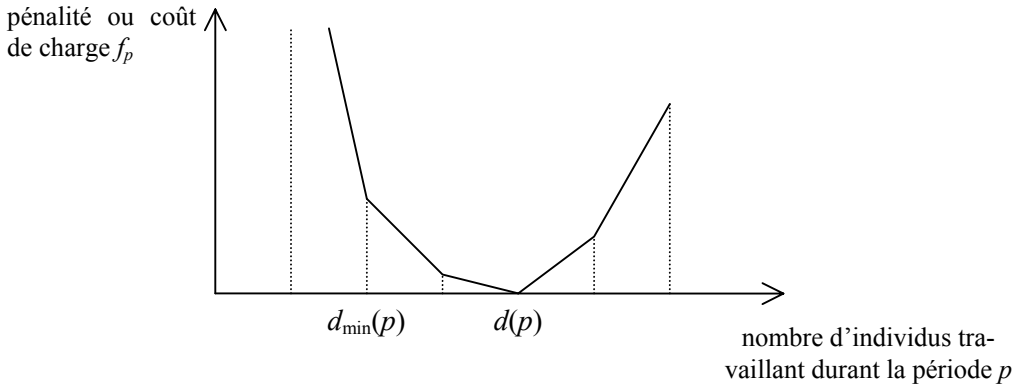


Figure 8 : Charge exprimée par une fonction de pénalité de charge (MILLER *et al.* [1976]).

$(d_{\min}(G, p), d(G, p), f_{G,p})$ triplet de pénalité de charge pour chaque période p de l'horizon de planification et pour chaque classe de personnel (voir Figure 8) ;

C_{soft} ensemble de contraintes de positionnement des repos qui, lorsqu'elles sont violées par un horaire, entraînent un coût ;

$ct(e, h)$ coût associé à chaque horaire de travail $h \in H(e)$ pour tout employé e. Ce coût dépend de trois autres paramètres qui sont :

$ct(e, c, h)$ est le coût associé au viol de la contrainte de positionnement des repos $c \in C_{soft}$ par l'horaire h de l'employé e ; $ca(e, c)$ est le coefficient d'aversion, accordé par l'employé à la violation de la contrainte c ; Ce poids est obtenu après une procédure d'interrogation de l'infirmier concerné sur la manière dont il perçoit la violation de cette contrainte par rapport aux autres contraintes et à un horaire de travail. $ia(e)$ est l'indice d'aversion de e pour rétablir une certaine forme d'égalité avec les autres employés, par rapport aux horaires affectés lors du précédent planning, et selon les préférences de l'employé. Ainsi, ce poids est d'autant plus élevé que la qualité des horaires passés est inférieure à celle des autres employés.

En définitive, le coût associé à l'horaire de travail h est obtenu par sommation, comme suit : $f(e, h) = ia(e) \cdot \sum_{c \in C_{soft}} ca(e,c) \cdot ct(e,c,h)$.

L'évaluation de la qualité d'un planning x est obtenue à partir d'une fonction coût comportant deux parties, un coût induit par la charge, et un coût social exprimant l'inverse d'un indicateur de satisfaction sociale :

f_c le coût total de pénalité de charge est obtenu à partir des deux types de triplets de pénalité de charge en sommant les fonctions de coût de pénalité de charge f_p et $f_{G,p}$:

$$f_c(x) = \sum_{p \in P} f_p\left(\sum_{e \in E} x(e,p)\right) + \sum_{p \in P}\sum_{G \in \mathcal{G}} f_{G,p}\left(\sum_{e \in E} x(e,p)\right),$$

avec $x(e,p) = \sum_{h \in H(e)} a(e,h,p) \cdot x(e,h)$ vaut 1 ssi l'horaire h affecté à l'employé e par le planning x compte la période p comme travaillée.

f_s le coût total social d'un planning correspond à la somme des coûts $f(e, h)$ sur les horaires sélectionnés par x.

Le problème à résoudre PEPTA(MPR) est alors d'affecter un horaire de travail à chaque employé, qui minimise une combinaison convexe du coût total de charge et du coût total social. Plus formellement nous avons :

PEPTA(MPR) : Étant donné $\lambda \in [0, 1]$, ainsi que le triplet $(I_{\text{PEPTA}}, f_c, f_s)$, les composantes étant définies comme spécifié précédemment, trouver une solution x vérifiant : $x \in S(I_{\text{PEPTA}})$, et minimisant la fonction $\lambda \cdot f_c(x) + (1 - \lambda) \cdot f_s(x)$.

2.2.3.5. Élaboration de Plannings Infirmiers selon WARNER [1976]

Le problème de WARNER PEPTA(W) est dédié à l'élaboration de plannings infirmiers, et considère une période correspondant à un couple étiquette-jour ($\Leftrightarrow P = EV \times J$). Comme PEPTA(MPR), WARNER utilise des classes de personnel mais de manière différente. De plus, il intègre les employés dans l'évaluation des horaires, lesquels peuvent exprimer l'importance qu'ils accordent à différentes situations locales pertinentes pouvant être retrouvées dans leurs horaires. Ainsi, PEPTA(W) considère les informations complémentaires suivantes :

$\mathcal{G}$ ensemble de classes de personnel effectuant une partition de E, ainsi qu'un ordre total $\geq$ sur cet ensemble exprimant l'inclusion des compétences d'une classe de personnel dans une autre : $G_1 \geq G_2 \Leftrightarrow$ les infirmiers de la classe G_1 peuvent effectuer les tâches de la classe G_2.
Les temps pleins et partiels ne constituent pas, dans ce problème, des propriétés définissant des classes de personnels différentes. Seules les compétences sont prises en compte.

$d_{\min}(G, p)$ demande minimale en infirmiers dont les compétences englobent celles de la classe de personnel G, durant la période p.

$d^+(G, p)$ maximum additionnel d'infirmiers dont les compétences englobent celles de la classe de personnel G, durant la période p. En d'autres termes, $d_{\min}(G, p) + d^+(G, p)$ forment une demande maximale.

$qs(G, p)$ valeur de qualité de service induite par la présence d'un infirmier de G supplémentaire durant la période p.

$im(G, p)$ coefficient de résistance à la pénurie en personnel de la classe G en la période p. $im(G, p)$ est un entier compris entre 0 et 5, qui croît avec l'augmentation de la résistance. Il est utilisé dans le cas où un sous-effectif serait inévitable.

$vs(e, h)$ valeur relative de l'horaire de travail h pour l'infirmier e qui est dérivée d'une combinaison de trois facteurs : (*a*) Les impressions de l'infirmier e envers les propriétés générales de l'horaire h ; (*b*) La conformité de l'horaire h aux désirs de jours de repos de l'infirmier e durant l'horizon de planification, et (*c*) La recherche d'une égalisation des horaires modifiés à cause d'imprévus.
Dans les exemples donnés dans l'article, ces paramètres sont des poids[28] qui sont sommés pour obtenir la valeur relative $vs(e, h)$.

[28] Le protocole d'acquisition de ces poids suit les grandes lignes suivantes :
– Chaque infirmier pondère les contraintes de positionnement des repos et des étiquettes qui n'ont pas été prises en compte dans la définition des horaires. La somme des poids doit être inférieure ou égale à une

WARNER utilise deux modèles pour obtenir un planning acceptable. Nous les détaillons ci-dessous :

❖ Cas où toutes les Demandes Minimales sont Satisfaites :

Le premier problème PEPTA(W1) contraint les solutions de $S(I_{\text{PEPTA}})$ à satisfaire les demandes minimales $d_{\min}(G, p)$.

L'évaluation de la qualité d'un planning x est obtenue à partir d'une fonction coût comportant deux parties, un coût induit par la charge, et un coût social exprimant l'inverse d'un indicateur de satisfaction sociale.

$qs(x)$ la qualité de service est obtenue en considérant :

$x_1(G, p)$ une variable d'écart correspondant au nombre d'infirmiers supplémentaires induit par le planning x par rapport à la demande minimale pour la classe G, durant la période p :

$$x_1(G,p) = \sum_{e \in \bigcup_{G^* \geq G} G^*} \sum_{h \in H(e)} a(e,h,p) \cdot x(e,h) - d_{\min}(G,p) \quad \text{pour tout} \begin{cases} G \in \mathcal{G} \\ p \in P = EV \times J \end{cases} \qquad (2\text{–}2)$$

$qs(x_1(G, p))$ une qualité de service par classe G et par période p est obtenue à partir de cette variable d'écart :

$$qs(x_1(G,p)) = \begin{cases} 0 & \text{si } x_1(G,p) = 0 \\ qs(G,p) & \text{si } 0 < x_1(G,p) \leq d^+(G,p) \\ qs(G,p) - (x_1(G,p) - d^+(G,p)) & \text{si } x_1(G,p) > d^+(G,p) \end{cases}$$

En définitive, la qualité de service $qs(x)$ correspond alors à la somme sur G et p de $qs(x_1(G, p))$.

$f_s(x)$ la satisfaction sociale agrégée d'une solution x est obtenue en effectuant la somme des valeurs relatives $vs(e, h)$ attribuées par e à l'horaire h qui lui est affecté par x :

$$f_s(x) = \sum_{e \in E} \sum_{h \in H(e)} vs(e,h) \cdot x(e,h)$$

Le problème à résoudre est alors un programme en nombre entiers défini par le système d'inégalités linéaires en nombres entiers $S(I_{\text{PEPTA}})$ auquel les contraintes (2–2) sont ajoutées et sur lequel la fonction de qualité agrégée $f_s + qs$ doit être maximisée.

❖ Cas où toutes les Demandes Minimales ne sont pas Satisfaites :

Dans ce cas – environ 20 à 30 % du temps – un second modèle PEPTA(W2) est construit, en introduisant une variable artificielle $x_2(G, p)$ par couple (G, p). Les contraintes de demande indexées par le couple (G, p), sont alors affaiblies en fonction du coefficient de résistance associé $im(G, p)$. Le nouveau système relaxé obtenu est utilisé pour résoudre un programme à fonction objective non linéaire définie par la somme sur G et p des fonctions :

---

valeur donnée. Les infirmiers à temps partiels reçoivent une quantité de poids inférieure à celle des temps pleins.

- Certaines situations de changement d'étiquettes sont pondérées par l'infirmière chef ;
- Les demandes de repos sont pondérées selon un autre système de poids par les infirmières demandeuses. Le pourcentage de poids non utilisé pour un planning est reporté pour le prochain horizon à planifier.
- Le cumul des différentes étiquettes est effectué à chaque planification, et utilisé comme outil pour mieux percevoir l'égalité et pour mieux évaluer les horaires de mauvaise qualité effectués dans le passé.

Cette méthode de quantification des préférences a pour but : « *to maintain 'fairness' in the amount of influence each nurse has in the decision process and to reduce favoritism, inadvertent or not, on the part of the head nurse* ».

$$f(x_2(G,p)) = \begin{cases} 0 & \text{si } x_2(G,p) = 0, \\ (10 - im(G,p))^{x_2(G,p)} & \text{si } x_2(G,p) > 0 \end{cases}.$$

Il existe bien d'autres exemples de PEPTA dans la littérature, ceux-ci étant étudiés directement ou indirectement par un autre PEPTP (en dimensionnant l'effectif ou en générant les horaires individuels (Figure 7)). En conclusion, le modèle PEPTA caractérise les problèmes qui admettent des employés et des horaires explicites. Les différences fondamentales interviennent au niveau de la modélisation des critères d'évaluation des plannings. Ainsi, plusieurs modèles sont apparus faisant un usage assez complexe de contraintes et de fonctions objectives. Il est souvent impossible de dissocier le protocole mis en place pour acquérir les informations, et les modèles informatiques utilisés.

2.2.4. Problèmes d'Élaboration de Plannings de Type Affectation Généralisée (PEPTAG)

Lorsque le nombre d'employés est connu et que les horaires sont donnés en compréhension, il est nécessaire de décrire l'ensemble des horaires admissibles par des contraintes en plus de la modélisation des critères d'évaluation des plannings. Ce type de problèmes, appelé PEPTAG pour problèmes d'élaboration de plannings de type affectation généralisé, regroupe des modèles très variés. La grande majorité des modèles basés sur les systèmes de contraintes se trouvent dans cette catégorie.

2.2.4.1. Éléments Caractéristiques

❖ **Le Cas le Plus Simple**

Dans le cadre le plus simple, les ensembles en extension connus et considérés sont E (employés) et P (périodes). La notion de période ici peut être remplacée par celle de jour J, ou tâche T. Pour concevoir un planning regroupant les horaires individuels construits, trois types de modèles sont possibles :

- Ceux utilisant les variables booléennes $x(e, p)$ associées aux couples employé-période : la valeur 1 indique que l'employé e est affecté à la période p par le planning x, alors que 0 informe du contraire ;
- Ceux faisant usage des variables $x(e)$ associées aux employés, le domaine réalisable étant un sous-ensemble des parties de P : $x(e) \in \mathcal{P}(P)$. En d'autres mots, un planning est décrit par l'ensemble des périodes de travail de chaque employé ;
- Ceux utilisant les variables $x(p)$ associées aux périodes, le domaine réalisable étant un sous-ensemble des parties de E : $D(x(p)) \subseteq \mathcal{P}(E)$. Un planning est décrit par l'ensemble des employés affectés à chaque période.

Généralement, la seule représentation opérationnelle est la première. Les deux autres modèles sont rarement utilisés tels quels[29], car la taille des domaines générés est exponentielle en la taille des données. Et, une représentation en compréhension des domaines (ensemble de sous-ensembles) est inconcevable dans le cas général. Par contre, d'autres possibilités ont été utilisées. Le second modèle est souvent ramené à un PEPTA, en énumérant tous les sous-ensembles de P qui respectent les contraintes horaires. Le troisième modèle est généralement utilisé lorsqu'un seul employé est à affecter par période (tâches), ou un nombre d'employés par période p très faible et fixe $d(p)$. Une variable $x(p, k)$ à valeurs dans E est alors associée à

[29] Voir cependant CHENG *et al*. [1997] pour une utilisation du troisième type de variables en élaboration de plannings infirmiers.

chaque $1 \leq k \leq d(p)$ et pour tout $p \in P$. Ces problèmes sont connus sous le nom de **employee scheduling** ou **personnel scheduling problems**.

❖ Les Cas Composés

L'utilisation d'étiquettes, de différentes classes d'employés, de différentes sous-unités au système autonome, etc. sont autant de facteurs qui diversifient les variables[30]. Dans le cas des compétences S, les deux variables les plus utilisées sont :

- les variables bivalentes $x(e, p, s)$; et
- les variables $x(e, p)$ à valeurs dans $S \cup \{\varepsilon_S\}$, avec ε_S exprimant que l'employé e n'est affecté à aucune compétence pour la période p ;

Les variables bivalentes sont plus adéquates pour représenter les contraintes sous forme linéaire (MIP), alors que le second modèle est privilégié pour les systèmes de contraintes.
Le nombre de modèles grandit de façon exponentielle avec le nombre d'entités pertinentes. Nous adoptons la notation (paramètres | valeurs) pour décrire la dichotomie des entités adoptée par un modèle. Par exemple, les variables d'un problème peuvent s'appuyer sur le quadruplet suivant (E, TV, J, S) formé des ensembles en extension d'employés E, de types de vacation TV, de jours J et de compétences S. Ces ensembles sont reliés par la relation d'affectation suivante : « un employé e en un jour donné j est affecté à un type de vacation tv pour effectuer des tâches nécessitant la compétence s ». Remarquons qu'il peut y avoir d'autres relations.

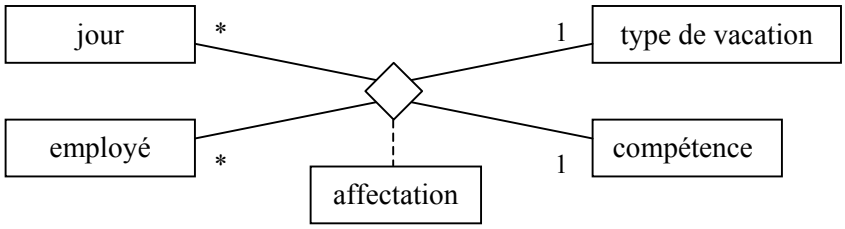


Figure 9 : Représentation UML de la relation d'affectation entre les jours, les employés, les étiquettes et les compétences.

Un planning x peut être décrit uniquement à partir des variables $x(e, j, tv) \in S \cup \{\varepsilon_S\}$, avec ε_S exprimant que l'employé e n'est pas affecté au type de vacation tv le jour j. Cette description repose donc sur la dichotomie (E, TV, J | S). Mais rien n'empêche d'utiliser simultanément ou exclusivement les variables $x(e, j) \in (TV \times S) \cup \{\varepsilon_{TV \times S}\}$, avec $\varepsilon_{TV \times S}$ exprimant par exemple que l'employé e est au repos le jour j. Les contraintes et critères sont alors formulés sur l'un et/ou l'autre des ensembles de variables ; certaines contraintes et/ou critères s'exprimant plus facilement dans un formalisme que dans un autre.

Une étude relationnelle des différentes entités permet de mieux comprendre et identifier les couples (paramètres | valeurs) qui n'engendrent pas de modèles dont la taille est exponentielle en celle des données dans le cas général. La relation d'affectation est appelée *channelling constraints* dans CHENG *et al*. [1997], [1999].

Les modèles peuvent reposer sur plusieurs relations d'affectations. Par exemple, une relation exprimant que « un employé e est au repos ou non durant la période p » et « e travaille selon la compétence s durant la période p ». Pour faciliter l'écriture en contraintes linéaires respectivement des contraintes de positionnement des repos et des contraintes de définition des vacations admissibles, les PCCG basés sur les modèles linéaires font un mixte des varia-

[30] Bien entendu, cette diversité n'est pas propre aux PEPTAG. Mais c'est dans ce cadre qu'elle est la plus marquée.

bles associées $x(e, p)$ et $x(e, p, s)$. Ces deux ensembles de variables sont reliés par la contrainte linéaire : $\sum_{s \in S} x(e, p, s) = x(e, p)$ pour tout $(e, p) \in E \times P$.

Les variables ajoutées peuvent être simplement des variables d'écart ou de 'linéarisation' du problème. Les contraintes entre groupes de variables peuvent être aussi des contraintes de connexion de niveaux, pour les modèles multi-niveaux (Par exemple planification journalière et mensuelle simultanée). Les contraintes de connexion de niveau peuvent être parfois très complexes.

2.2.4.2. Quelques Modèles de la Littérature

❖ Les Variables

Utilisons la dichotomie effectuée par les variables pour trier les différents problèmes de la littérature.

La quasi-totalité des modèles PEPTAG modélisés par les systèmes de contraintes se limitent aux problèmes d'élaboration de grilles étiquetées CASEAU *et al.* [1995], HEUS [1996], MIETUS [1994], OLDENKAMP [1996], PARTOUCHE [1998], ABDENNADHER & SCHLENKER [1999], MUSLIU [2001], MEYER AUF'M HOFE [2001] et CHAN [2002]. Ces problèmes supposent connus les ensembles d'employés E, de jours J et d'étiquettes EV en extension. De plus, ils sont tous basés sur la séparation paramètre-valeur $(E, J \mid EV)$.

CHAN [2002], utilise en plus un ensemble de compétences S en extension, avec des employés mono-compétents. La mono-compétence de travailleurs et la particularité des contraintes permet de traiter ce problème comme $\mathrm{card}(S)$ problèmes $(E, J \mid EV)$ indépendants. CHENG *et al.* [1997], [1999] font usage d'un couple de modèles $(EV, J \mid \mathcal{P}(E))$ et $(E, J \mid EV)$. JASZKIEWICZ [1997], [2001] lui aussi utilise un modèle mixte $(E, J \mid EV)$ et $(E, J, EV \mid \{0, 1\})$.

La plupart des PEPTAG modélisés en MIP sont cycliques (utilisation d'une contrainte de cyclicité). Ils sont présentés au paragraphe suivant. Parmi les modèles non cycliques, citons les PPR présentés par FERLAND [1998] et BILLIONET [1999] utilisant la dichotomie $(E, J \mid \{0, 1\})$, et GASCON *et al.* [2000] traitant d'un problème 'multi-unité'. Détaillons leur modèle :

Le but est de résoudre un problème de positionnement des repos d'infirmiers appartenant à une équipe volante : Les hôpitaux sont organisés de différentes manières. Le système hospitalier considéré par GASCON *et al*. [1999], distribue les infirmiers en quatre groupes distincts : (*a*) les infirmiers liés à une unité de soin particulière, (*b*) les infirmiers appartenant à une unité volante localisée géographiquement (*area flying squad*), c'est-à-dire des infirmiers délocalisés sur un ensemble d'unités de soins, (*c*) les infirmiers appartenant à l'unité volante générale : ils peuvent être affectés à n'importe quelle unité, et (*d*) les infirmiers appartenant à une liste de rappel : ils sont requis lorsque les infirmiers des groupes précédents ne peuvent pas ensemble satisfaire les besoins journaliers. Cette organisation suggère directement un processus séquentiel bien défini de résolution des problèmes d'élaboration de plannings infirmiers : La planification est réalisée en affectant en priorité les infirmiers appartenant à l'unité la plus spécialisée (l'unité de soin), puis ceux de l'unité la moins spécialisée (les infirmiers de la liste de rappel) selon les besoins non satisfaits par les groupes précédents. GASCON *et al*. utilisent 5 types de variables différentes pour décrire un planning x. Si U est l'ensemble des sous-unités de l'unité volante, et $\{W_i, W_f\}$ une bipartition de l'horizon de planning en deux séquences de semaines consécutives, alors :

- La variable bivalente $x(e, j, u)$ vaut 1 ssi le travailleur e a été affecté le jour j à l'unité u.
- La variable bivalente $x(e, j)$ vaut 1 ssi le travailleur e travaille le jour j.
- La variable bivalente $x(e, u, WP)$ vaut 1 ssi le travailleur e travaille dans l'unité u durant la période $WP \in \{W_i, W_f\}$ de l'horizon de planning.

– La variable bivalente $x(e, u)$ vaut 1 ssi le travailleur e travaille dans l'unité u moins de jours qu'un nombre de jours fixé, ou au moins deux semaines lors du précédent horizon.

❖ Les Contraintes et Critères

Concernant les contraintes, elles portent sur le positionnement des repos et des étiquettes. Certains modèles intègrent des souhaits individuels du personnel comme contraintes dures, avant la planification et après validation par des acteurs tiers. Les souhaits sont tous formulés comme une demande de congés ou de travail selon certains types de vacation. Nous ne rentrerons pas plus dans les détails des contraintes, leur nombre, leur complexité et leur spécificité étant très grands. Signalons le document de BELDICEANU [2000*a*] établissant une liste de contraintes utilisables en élaboration d'horaires de travail proposés dans les logiciels industriels tels `Chip` et `IlogSolver`. Dans le même cadre, RÉGIN [1996] proposa la contrainte de cardinalité généralisée (voir Annexe A1).

La qualité de service et les coûts sont généralement modélisés par une table de besoins à couvrir exactement.

Les critères sociaux sont modélisés de deux manières différentes. La première (JASZKIEWICZ [1997], [2001], OLDENKAMP [1996], GASCON *et al*. [1999], etc.) consiste à utiliser un ensemble assez restreint – voir une seule – fonctions objectives, dont le sens n'est pas toujours très clair, et où il faut une période d'adaptation au planificateur pour comprendre un certain nombre de subtilités que renferme chaque critère. Les repères sont perdus si des instances relativement différentes des usuelles sont utilisées. Ces défauts sont d'ailleurs reconnus par certains auteurs. Ces fonctions objectives sont obtenues après plusieurs combinaisons partielles, de manière hiérarchique, d'indicateurs et/ou contraintes souples.
Une seconde solution (CASEAU *et al.* [1995], HEUS [1996], PARTOUCHE [1998], CHAN [2002], etc.) se positionne à un niveau d'agrégation plus faible : Un grand nombre d'indicateurs (ou compteurs) est utilisé, chacun s'interprétant très simplement comme un nombre de contraintes souples d'un type donné non respectées par un planning. Par exemple, le nombre de repos isolés pour l'individu e est de $\rho(x, e)$ pour le planning x. C'est-à-dire que $\rho(x, e)$ contraintes de repos isolé interdit, sont violées pour e dans x. La résolution se déroule alors en usant de techniques de la programmation multiobjective (CHARNES & COOPER [1977]), certains privilégiant plus que d'autres l'interaction avec les utilisateurs.

Remarquons qu'il existe différentes relations entre les contraintes et les critères d'un modèle. Par exemple (JASZKIEWICZ [1997], [2001]) la contrainte de borne sur le nombre de jours consécutifs de travail représente une sorte de niveau de qualité minimal pour la fonction objective de minimisation de la moyenne des déviations autour d'une valeur idéale, des séquences de jours de travail. De même, la charge formulée en termes de besoin minimum et maximum en chaque période, correspond à des niveaux de qualité minimum respectivement pour la qualité de service et le coût en ces périodes.

2.2.4.3. Problèmes Cycliques

Les problèmes cycliques classiques (par exemple BAKER [1976], PARTOUCHE [1998]) consistent à élaborer des grilles cycliques. Leurs instances se basent sur un ensemble des jours de semaines $JS = \{\text{lundi}, \ldots, \text{dimanche}\}$, un ensemble de semaines consécutives W et un ensemble d'étiquettes EV. Les variables principales sont $(JS, W, EV \mid \{0, 1\})$ et les variables auxiliaires $(JS, W \mid \{0, 1\})$. Le nombre d'employés est inconnu, mais il respecte une contrainte de cyclicité dont les caractéristiques sont les suivantes :

– le nombre de travailleurs $\text{card}(E)$ est un multiple de $\text{card}(W)$,
– les employés sont répartis en $\text{card}(W)$ équipes de $\text{card}(E) / \text{card}(W)$ employés ayant les mêmes horaires de travail,
– chaque équipe est affectée à une semaine,

– un ordre d'enchaînement des semaines (formant un circuit élémentaire de longueur card(W) sur W), appelé aussi **roulement**, permet d'obtenir les horaires de travail de chaque travailleur. Le roulement indique qu'une équipe affectée à la semaine w_1 effectuera ensuite la semaine w_2 qui le suit dans le roulement, et ainsi de suite.

Par exemple, le Tableau 1 est une grille cyclique, affectant une équipe par ligne, et l'équipe affectée à la ligne l une semaine donnée, avec $l <$ card(W), effectuera la ligne $l + 1$ la semaine suivante, et la ligne 1 si $l =$ card(W).

Les problèmes classiques se différencient les uns des autres par les contraintes supplémentaires à satisfaire qui sont des contraintes de positionnement des repos et de positionnement des étiquettes.

Ces dernières années, d'autres problèmes cycliques moins classiques ont vu le jour. Par exemple LAPORTE [1999] considère une contrainte de cyclicité admettant plus de souplesse dans l'ordre d'enchaînement des semaines. Pour cela, il admet un planning multi-grille GR partitionnant W : $W = \{W(gr_1), \ldots, W(gr_q)\}$ avec $W(gr_i)$ le nombre de semaines (périodicité) de la grille $gr_i \in GR$. PARTOUCHE [1998, § 8.5] utilise aussi un problème 'multi-grille', le nombre de grilles étant connu ainsi que leur périodicité. CHAN & WEIL [2000] et CHAN [2002] assouplissent la périodicité des roulements qui n'est plus un multiple de 7 jours, et établissent un algorithme de transformation d'un roulement de périodicité p_1 en un de périodicité p_2 conservant la charge journalière.

2.2.4.4. L'Auto-Planification

Ce type de planification a été développé dans certains services de soins hospitaliers[31]. L'**auto-planification** (*Self-scheduling*) est une méthode interactive d'élaboration de plannings de travail au cours de laquelle l'infirmière gestionnaire (*nurse manager*) détermine les besoins en personnel par jour et par vacation, et où les infirmières (*nursing staff*) ordonnancent elles-mêmes leurs horaires de manière à satisfaire ces besoins. Plusieurs autres variantes sont possibles.

Protocole : Pour éviter des abus, un protocole précis est mis en place pour aboutir à un planning final. Tout d'abord, la charge en tant que compromis entre les coûts et la qualité de service est préalablement établie puis validée par les gestionnaires de l'unité. La couverture exacte est l'assurance d'une qualité de service et d'un coût presque constant sur l'ensemble des plannings réalisables[32]. La force de travail (nombre total d'heures de travail à effectuer par l'ensemble du personnel) est donc connue, et seule la répartition du personnel est à déterminer. Les infirmières se réunissent alors entre elles pour négocier leurs horaires. Ce processus est généralement couronné de succès. Le planning négocié doit être remis à l'infirmière gestionnaire, avant une date limite se situant plusieurs jours avant la date effective de réalisation, pour qu'il soit validé ou remis en question. Lors de situations conflictuelles où aucun consensus n'a pu être trouvé, le recours à un médiateur (généralement l'infirmière gestionnaire) est effectué.

Avantages : Les avantages d'une telle élaboration de plannings sont indéniables. (*a*) Pour le personnel, elle lui offre un meilleur contrôle de ses horaires de travail (liberté de choix sur la manière de gérer sa vie). Renforce la cohésion du groupe, le sentiment d'appartenance au groupe. En résumé, l'auto-planification améliore bon nombre de critères de qualité de vie au travail (QwL) : la satisfaction au travail, amélioration de la productivité, de la motivation, une meilleure adéquation entre la vie professionnelle et la vie extra-professionnelle. (*b*) Pour les gestionnaires, une meilleure compréhension des besoins de l'unité de la part du personnel, est

[31] Voir KRISHER [2000]. MIETUS [1994] et OLDENKAMP [1996] fournissent un ensemble de références plus axés sur l'utilisation de l'ordinateur dans ce type de planification.
[32] Les indicateurs sur lesquels prennent appui ces critères, ne tiennent pas compte des effets rétroactifs.

une réelle solution à l'absentéisme. Les infirmières s'approprient en grande partie les besoins de leur unité, en réalisant leurs horaires de manière à être sûres que toutes les demandes sont satisfaites (KRISHER [2000]). Par conséquent une meilleure implication au travail[33]. Un autre avantage, qui n'est pas des moindres au vu de la pénurie d'infirmières existant dans grand nombre de pays (EU, Australie, France, etc. (KRISHER [2000], BONMATI [1998])), c'est un gage de qualité lors de recrutements, et de rétention du personnel (KRISHER [2000]). (*c*) Pour les clients, une meilleure implication au travail est synonyme d'une amélioration de la qualité des soins. (*d*) Bien qu'elle ait été développée dans le secteur santé, cette planification peut s'ouvrir à d'autres secteurs d'activité comme les services postiers, banquiers, pénitenciers, etc.

Inconvénients : Plusieurs reproches ont été adressés à ce type de gestion. (*a*) Du temps décompté souvent sur les heures de travail doit être réservé à la négociation. (*b*) Une telle planification n'est adaptée qu'à des unités cibles de petite taille (≤ 20 employés). (*c*) Cette planification semble être inadaptée aux problèmes impersonnels, ou impliquant d'autres ressources à planifier simultanément. (*d*) Les besoins doivent être établis avec précision, pour une couverture exacte.

Rôle de l'ordinateur dans une auto-planification : L'auto-planification est d'abord une nouvelle forme de gestion, accordant plus de responsabilités au personnel. Elle établit une mini-révolution dans le secteur de la santé souvent si hiérarchisé (voir BONMATI [1998]), couplée à une forme de gestion non participative, souvent préjudiciable à un bon climat social. Elle nécessite aussi de revoir le rôle occupé par l'ordinateur dans le processus d'élaboration de ces plannings. L'automatisation complète de la planification n'est plus à l'ordre du jour. Les logiciels conçus jusqu'à maintenant sont surtout axés sur l'IHM. Pour le moment, la proportion d'unités de soins usant de l'auto-planification est très faible (moins de 5 %). Cette tâche reste toujours très centralisée. Une réelle évolution des mentalités (modes de gestion mais aussi de la conception de logiciels d'élaboration de plannings) reste encore à faire. Nous verrons au § 3.2.2 qu'une véritable aide à la négociation peut être portée par l'ordinateur dans ce cadre.

2.2.5. Problèmes d'Élaboration de Plannings de Type Recouvrement d'Ensemble Généralisé (PEPTREG)

Les problèmes d'élaboration de plannings de type recouvrement d'ensemble généralisé (PEPTREG) supposent que l'effectif est inconnu, contrairement à l'ensemble des horaires de travail donnés en extension. Ils se distinguent par des caractéristiques contractuelles différentes et des manières variées de calculer les coûts, de représenter la qualité de service et la satisfaction du personnel. Le nom de ce type de problèmes vient du fait que la plupart des modèles qui s'y rattachent définissent la qualité de service comme un recouvrement de la demande minimale en chaque période de travail par un ensemble d'horaires. Seul le sur-effectif est admis. Mais nous regroupons sous cette dénomination les problèmes dont la charge doit être couverte exactement, ou peut admettre du sous-effectif. Les PEPTREG mettent généralement l'accent sur la charge (qualité de service + coûts). La plupart de ces problèmes reposent tous sur le modèle standard qu'est le **problème de recouvrement d'ensemble généralisé** qui sera présenté au § 2.2.5.1. Puis nous déboucherons au § 2.2.5.2 sur différents problèmes de ce type présenté dans la littérature.

2.2.5.1. Le Problème de Recouvrement d'Ensemble Généralisé (GSCP)

Le **problème de recouvrement d'ensemble généralisé** (GSCP : *Generalized Set Covering Problem*) fut introduit en RO en 1954 par DANTZIG [1954], marquant par la même occasion

[33] L'implication au travail est l'un des thèmes qui sont chers aux organisateurs (THEVENET [1993]).

les débuts de l'élaboration de plannings comme type de problème pouvant être traité par des systèmes informatiques. Le GSCP élabore un planning de personnel à partir de la connaissance explicite des horaires individuels, de la charge et des coûts horaires. Formellement, une instance I_{GSCP} d'un GSCP est formée de :

P ensemble des périodes,

H ensemble des horaires de travail,

$d_{\min}(p)$ demande minimale en chaque période p de P, c'est-à-dire, le nombre nécessaire de travailleurs durant la période p pour effectuer un niveau de service désiré minimal,

$a(h, p)$ valeur booléenne égale à 1 si la période p est une période de travail pour l'horaire h, et 0 sinon ;

$c(h)$ coût au travailleur de l'horaire h, $\forall\ h \in H$.

Un planning est décrit par l'ensemble des variables entières positives $x(h)\ \forall\ h \in H$, la variable $x(h)$ étant définie comme le nombre de travailleurs à affecter à l'horaire h. Le problème se formule alors comme un programme linéaire en nombres entiers comme suit :

GSCP : Étant donnée une instance I_{GSCP}, trouver une solution x, si une telle solution existe, minimisant la fonction de coût linéaire $\sum_{h \in H} c(h) \cdot x(h)$, et satisfaisant les contraintes linéaires et d'intégrité suivantes :

$$\begin{cases} \sum_{h \in H} a(h, p) \cdot x(h) \geq d_{\min}(p) & \text{pour tout } p \in P \\ x(h) \in N & \text{pour tout } h \in H. \end{cases}$$

Sinon retourner "non".

L'inégalité exprime que le nombre d'employés travaillant lors d'une période, doit couvrir la demande de cette période.

Ce problème est l'un des plus étudiés en RO raison des différents problèmes – pas forcément des PEPTP – pouvant être modélisés de cette manière. PARTOUCHE [1998] répertorie un ensemble d'articles traitant du GSCP en élaboration de plannings.

2.2.5.2. Quelques Problèmes de Type Recouvrement d'Ensemble Généralisé

Plusieurs extensions, spécialisations et variantes de ce problème existent dans la littérature. PARTOUCHE [1998, § 8.3] considère une généralisation à plusieurs classes de personnel $\mathcal{G}$. Les classes se différencient uniquement par la durée légale de travail sur l'horizon étudié. Soit D_{LTH} ces différentes durées : $\mathcal{G} = \{G^{\langle\delta\rangle}$ tel que : $\delta \in D_{LTH}\}$. Chaque classe $G^{\langle\delta\rangle}$ est associée à un quintuplet ($H(\delta)$, $lbv(\delta)$, $ubv(\delta)$, $lbd(\delta)$, $ubd(\delta)$), où $H(\delta)$ est un ensemble en extension d'horaires de travail admissibles, $lbv(\delta)$ et $ubv(\delta)$ des bornes sur le nombre total d'horaires à sélectionner pour la classe $G^{\langle\delta\rangle}$, et $lbd(\delta)$, $ubd(\delta)$ des bornes sur la durée totale de travail sur l'horizon P de la classe $G^{\langle\delta\rangle}$. Elle suppose aussi donnée la durée totale de travail $\text{durée}(h)$ de chaque horaire h. Alors son problème est défini en considérant le programme linéaire en nombres entiers GSCP auquel sont ajoutées les contraintes de bornes suivantes :

$$lbv(\delta) \leq \sum_{h \in H(\delta)} x(h) \leq ubv(\delta) \quad \text{et} \quad lbd(\delta) \leq \sum_{h \in H(\delta)} \text{durée}(h) \cdot x(h) \leq ubd(\delta) \quad \text{pour tout } \delta \in D_{LTH}.$$

PARTOUCHE appelle ce problème PCV-contraint, mais il peut être appliqué en dehors du cadre du PCV en élaboration de plannings.

BRUSCO et JOHNS [1996] considèrent aussi un problème portant sur plusieurs classes de personnel différenciées par la durée légale de travail sur l'horizon, ainsi que la performance d'un employé induite par l'horaire de travail considéré. Cette performance permet d'exprimer des préférences entre les horaires de travail, la préférence allant pour l'horaire ayant la per-

formance la plus élevée. Ce problème qui est un PCT, est appelé par BRUSCO & JOHNS *General Set-Covering Formulations* (GSCF).

CAI & LI [2000] eux considèrent une généralisation du GSCP vers plusieurs critères et plusieurs classes de personnel (*staff scheduling problem*). Le premier critère correspond à la minimisation de la somme des coûts horaires $c(h)$; le second maximise la somme des sur-effectifs, et le dernier minimise la somme des variances des sous- et sur-effectifs normalisée par la demande. Le second critère a pour fonction de réduire le risque de sous-estimation de la demande réelle qui peut arriver à cause des imprécisions de la prévision de la charge. Le troisième critère permet de mieux équilibrer la répartition de l'effectif le long de la courbe de charge, lequel est souvent vu comme un facteur permettant d'obtenir une solution équitable.

RUSHMEIER *et al.* [1995] utilisent le recouvrement exact (connu sous le nom de **problème de partitionnement d'ensemble généralisé** (*generalized set-partitioning problem*)), lequel est obtenu en remplaçant l'inégalité de la contrainte de charge du système de contraintes définissant GSCP par l'égalité. Ils effectuent une synthèse des problèmes de rotation d'équipages PCR (*crew scheduling*) tous basés sur un recouvrement exact de la charge formulée par une équipe par tronçon de vol (les tronçons et les rotations correspondent respectivement aux tâches à effectuer et aux horaires de travail). Il est intéressant de constater la complexité sous-jacente au coût horaire $c(h)$ de ce type de problème : il correspond aux dépenses occasionnées par la rotation incluant les coûts associés à la durée passée loin du domicile, la quantité d'heures de vol dans la rotation, les charges d'hébergement, les frais de transport au sol et des transports 'gratuits' d'équipage (transport d'un équipage sur un avion où il ne travaille pas). Les horaires sont généralement obtenus par énumération (§ 2.2.7).

2.2.6. Problèmes d'Élaboration de Planning de Type Implicite (PEPTI)

Les problèmes d'élaboration de planning de type implicite (PEPTI) sont caractérisés à la fois par un nombre d'employés inconnu et des horaires individuels qui ne sont pas explicitement connus. Ils ont fait leur apparition d'abord dans les problèmes de construction de vacations (PCV), mais les modèles utilisés sont aussi valables pour les autres types de problèmes d'élaboration de plannings présentés au § 2.2.2.1.

La quantité d'information donnée est très importante : les périodes, les coûts, la charge, les contraintes de définition des horaires et les classes de personnel. Le but est alors de dimensionner l'effectif et construire des horaires définissant un planning de meilleur compromis.

Il existe un grand nombre de PEPTI répertoriés dans la littérature. Dans le cadre du PCV nous avons : SEGAL [1974], MOONDRA [1976], BECHTOLD & JACOBS [1990], AYKIN [1996], THOMPSON [1995], PARTOUCHE [1998], JACQUET-LAGRÈZE et *al*. [1998] et AYKIN [2000], parmi d'autres. Dans le cadre du PCCG nous avons : BILLIONET [1999], BRUSCO & JACOBS [2000], LI & LI [2000], BEAUMONT [1999], NARASIMHAN [2000] pour ne citer que les plus récents, et pour le PGCG il y a les travaux de PARTOUCHE [1998, chap. 8].

Nous présentons ci-dessous un modèle assez général introduit par AYKIN [1996] pour mieux se rendre compte de la complexité sous-jacente :

2.2.6.1. Le Modèle Implicite d'AYKIN [1996]

Le PEPTI introduit par AYKIN [1996] est un modèle linéaire entier qui suppose connus :

P l'ensemble des périodes,

K_H un ensemble d'intervalles de dates – appelés aussi fenêtres de temps – pour lesquels il existe au moins un horaire de travail ayant exactement sa date de début et de fin. Chaque intervalle de date k de K_H, possède les attributs suivants : Une date de début : $\text{début}(k)$, une date de fin : $\text{fin}(k)$ et une durée : $\text{amplitude}(k)$. Ces attributs sont liés par la relation : $\text{fin}(k) - \text{début}(k) = \text{amplitude}(k) > 0$

L_H un ensemble de couples de (période, entier positif) (p, i) de $P \times \mathbb{N}^*$, pour lequel il existe au moins une vacation comportant au moins i coupures (pauses, repos) et où p est une période de repos appartenant à la i^e coupure[34].

$c(k)$ coût par personne et par horaire de travail vérifiant k,

Un planning est décrit par l'ensemble des variables entières positives $x(k)$ et $x(k, p, i)$ pour tout $k \in K_H$ et $(p, i) \in L_H$. Les variables entières positives $x(k)$ et $x(k, p, i)$ sont définies comme le nombre de travailleurs affectés aux sous-ensembles d'horaires de travail, respectivement $H(k)$ et $H(k, p, i)$. $H(k)$ est l'ensemble des horaires de travail ayant k comme intervalle de dates ; et $H(k, p, i)$ est l'ensemble des horaires de travail ayant k comme intervalle de dates contenant au moins i coupures, la i^e incluant la période p.

Le but du problème est alors d'identifier un planning x, formé de toutes les variables entières positives $x(k)$ et $x(k,p,i)$, vérifiant le critère et les contraintes suivantes :

$$\text{Minimiser } \sum_{k \in K_H} c(k) \cdot x(k)$$

$$\text{Sous les contraintes : } \begin{cases} \sum\limits_{\substack{k \in K_H \\ \text{tel que } p \subseteq k}} x(k) - \sum\limits_{\substack{k \in K_H \\ \text{tel que } p \subseteq k}} \sum\limits_{\substack{i \text{ tel que} \\ (p,i) \in L_H}} x(k,p,i) \geq d(p) \quad \text{pour tout } p \in P \\ x(k) - \sum\limits_{\substack{p \in P \\ \text{tel que} \\ (p,i) \in L_H \text{ et } p \subseteq k}} x(k,p,i) = 0 \quad \text{pour tout } k \in K_H \text{ et } i \in \mathbb{N}^* \\ x(k), x(k,p,i) \in \mathbb{N} \quad \text{pour tout } k \in K_H \text{ et } (p,i) \in L_H. \end{cases}$$

Remarquons que par l'assertion " $p \subseteq k$ ", nous nous référons à l'intervalle de dates sous-jacent à p lequel est inclus dans l'intervalle de dates k.

Le critère de coût est égal à la somme sur k, du nombre de travailleurs affectés à un horaire de $H(k)$ pondéré par le coût unitaire $c(k)$. Remarquons que chaque horaire de $H(k)$ a le même coût par travailleur $c(k)$.

L'inégalité spécifie que la demande pour une période p doit être inférieure ou égale au nombre de personnes travaillant durant p ; Ce dernier nombre est égal au nombre de travailleurs ayant un horaire contenant p, moins le nombre de travailleurs étant en pause (ou repos) durant cette période.

L'égalité spécifie que le nombre de travailleurs dont l'intervalle de dates sous-jacent à leur horaire est k, est égal au nombre de travailleurs prenant leur i^e pause durant des périodes incluses dans k. Cette égalité suppose que, pour un intervalle de temps k donné, les différents horaires de $H(k)$ ont un même nombre de pauses. Ce qui est souvent le cas dans le cadre du PCV, mais qui n'est plus valable dans un cadre plus général.

Cet exemple montre quelques difficultés qui peuvent être rencontrées lors de la modélisation des problèmes de type PEPTI. De plus, les problèmes sont très divers car ils reposent sur des données identifiant des sous-ensembles d'horaires très différents.

2.2.7. Génération d'Horaires de Travail

Les problèmes de génération d'horaires de travail permettent de passer des problèmes d'élaboration de plannings de personnel dont les horaires sont donnés en compréhension (PEPTAG et PEPTI), vers les mêmes problèmes où les horaires sont donnés explicitement (PEPTA et PEPTREG). Ces problèmes utilisent les contraintes de description des horaires de travail admissibles : Contraintes de définition des vacations admissibles, de tâches, de positionnement des repos et des étiquettes.

[34] Les coupures étant rangées (numérotées) dans l'ordre croissant de leur apparition dans le temps.

Remarquons qu'ici encore, l'ensemble des périodes P peut être remplacé par J, $EV \times J$, etc.
Deux cas de génération sont utilisés dans la littérature :

- totale : On parle alors d'énumération des horaires individuels (*matrix-generation*) ; ou
- partielle : On parle alors de sélection d'un sous-ensemble d'horaires candidats (*column generation*).

Illustrons les deux formes de génération par des exemples :

2.2.7.1. Énumération des Horaires Individuels

Le but est d'énumérer l'ensemble des horaires de travail de manière à pouvoir ensuite l'utiliser dans un problème à horaires explicites. Une définition générique est la suivante :

Enum(SUBSET) : Étant donné un ensemble P, et un ensemble de contraintes $C(P)$ spécifiant les éléments admissibles de $\mathcal{P}(P)$, retourner la liste des parties de P réalisables.

Prenons le modèle de PARTOUCHE [1998, § 5.1] qui consiste à générer l'ensemble des vacations réalisables d'une journée. Une instance d'un tel problème est composée de :

P ensemble totalement ordonné des périodes de travail numérotées par ordre croissant : pour tout $i \in \{1, \ldots, \text{card}(P) - 1\}$ $\quad \text{début}(p_{i+1}) \geq \text{fin}(p_i)$

PA ensemble de types pauses

et d'un ensemble d'attributs permettant d'exprimer les contraintes de bornes :

la, ua : durées minimum et maximum d'une vacation (en nombre de périodes)

ld, ud : périodes de début au plus tôt et au plus tard d'une vacation

durée(pa) :durée d'une pause de type $pa \in PA$ (en nombre de périodes)

fixe(pa) : booléen indiquant si une pause de type pa est absolue (ses caractéristiques ne dépendent pas de la vacation à laquelle elle appartient) ou relative (dépendante)

$ld(pa)$, $ud(pa)$: périodes de début au plus tôt et au plus tard d'une pause de type pa absolue (fixe(pa) = oui)

$ltavp(pa)$, $utavp(pa)$: nombre minimum et maximum de périodes de travail avant une pause de type pa

$ltapp(pa)$, $utapp(pa)$: nombre minimum et maximum de périodes de travail après une pause de type pa

$apas$: nombre de périodes séparant deux amplitudes possibles pour une vacation

$dpas$: nombre de périodes séparant l'heure de début de deux vacations

$papas$: nombre de périodes séparant le début de deux positions alternatives d'une pause au sein d'une vacation

Le problème étudié est polynomial en la taille des données. Pour quelques résultats de dénombrement des solutions réalisables, voir PARTOUCHE [1998, § 5.1.2].

2.2.7.2. Génération Partielle des Horaires

La complexité des règles de droits de certains secteurs, et la taille de l'ensemble des périodes (ou tâches) interdisent la génération exhaustive de tous les horaires réalisables (limitation de la mémoire des ordinateurs). A titre d'exemple, pour un problème d'élaboration de plannings du personnel naviguant de compagnie aérienne (RUSHMEIER *et al.* [1995]), formé de 1000 tronçons de vol, il existe quelques milliards de rotations réalisables et les plannings de bonne qualité comportent environ 120 rotations.

Ainsi, une alternative à l'énumération totale est la mise en place de stratégies d'énumération partielle de l'ensemble des horaires de travail. En voici quelques unes (RUSHMEIER *et al.* [1995]) :

1e stratégie : (*a*) choisir aléatoirement un sous-ensemble de périodes de cardinal donné, (*b*) générer tous les horaires réalisables possibles contenant les périodes de ce sous-ensemble.

2nde stratégie : (*a*) générer un sous-ensemble d'horaires réalisables représentatif de l'ensemble des périodes (pour chaque période il existe au moins un horaire qui la contient), (*b*) utiliser une génération aléatoire qui utilise des règles d'échantillonnage favorisant les horaires de « faible coût ».

D'autres stratégies de génération partielle sont basées sur l'utilisation d'un graphe orienté acyclique ayant un sommet initial (source) et final (puits). Chaque chemin joignant la source et le puits représente un horaire de travail.

Le problème de stratégie défini par JAUMARD *et al.* [1998] pour l'élaboration de plannings infirmiers (voir aussi HANSEN *et al.* [1998] dans le cas bicritère), est un **problème de plus court chemin contraint par des ressources**. Il s'exprime de façon informelle comme suit :

Étant donné un planning x, un travailleur e, notons $h(x, e)$ l'horaire individuel de travail du travailleur e pour x. Le but est alors de trouver un autre horaire h satisfaisant les contraintes de droit du travail de e, respectant un ensemble de contraintes de validité si $h(x, e)$ est remplacé par h dans x pour obtenir un nouveau planning x', et minimisant un coût $c(h)$.

Les contraintes de validité sont exprimées par la topologie du graphe orienté acyclique $G_{x,e}$ associé au couple (x, e). Les contraintes de droit du travail caractérisant les horaires réalisables, sont représentées par des ressources en quantités bornées sur les sommets du graphe. Ainsi, 7 ressources différentes sont considérées dans le problème de JAUMARD *et al.* [1998] : la durée hebdomadaire de travail, le nombre de week-ends de repos, le nombre de vacations consécutives d'un type donné (matin, soit, nuit), le nombre de jours travaillés consécutifs, la proportion de chaque type de vacation (définissant deux ressources distinctes), et le nombre de jours de repos.

Sachant que chaque arc consomme une partie des ressources, le but est de trouver un chemin h de la source au puits (= un horaire), restant dans les limites de chaque ressource, et minimisant le coût $c(h)$. Le coût $c(h)$ correspond au salaire de e induit par h, auquel est soustrait le niveau de satisfaction des préférences de e atteint par h, ainsi qu'un troisième terme correspondant au désir du niveau de compétence de e dans le nouveau planning x', calculé à partir de x.

Remarquons ici que toutes les préférences sont exprimées numériquement, et qu'il est possible de les comparer et les cumuler à des valeurs d'autres dimensions comme le salaire.

2.2.8. Problèmes de Dimensionnement d'Effectif

Les problèmes de dimensionnement consistent à évaluer l'effectif d'une solution de bonne qualité d'un problème d'élaboration de plannings de personnel. Le dimensionnement intervient à plusieurs niveaux de la planification à moyen terme (planification annuelle), et/ou court terme. A court terme, le dimensionnement concerne le personnel intérimaire, non permanent, d'unités fédératrices formées d'équipes volantes, pour combler les pics, fortes fluctuations de la charge et événements exceptionnels. Dans le moyen terme, elle concerne le personnel à contrat à durée déterminée (CDD), et débouche directement sur des problèmes plus généraux de gestion de ressources humaines comme le recrutement ou la gestion prévisionnelle des flux d'employés (WIJNGAARD [1983]).

Dans notre cadre, les problèmes de dimensionnement permettent de passer des problèmes d'élaboration de planning de travail de personnel dont le nombre d'employés est donné en compréhension (PEPTI, PEPTREG), vers les problèmes où ce nombre est donné explicitement (PEPTA, PEPTAG).

Plus formellement, le problème de dimensionnement associé à un sous-problème $\Pi \subseteq$ PEPTI (resp. $\Pi \subseteq$ PEPTREG), d'élaboration de planning de personnel (PEPTP), est un problème combinatoire DIM(Π) associant à toute instance I de Π, l'ensemble des valeurs pouvant être prises par l'effectif du personnel d'un planning réalisable. Si l'on note $E(x)$ la population réalisant une tâche dans le planning x, alors :

DIM(Π) : Étant donné une instance I du sous-problème d'élaboration de planning de personnel Π, retourner un entier $\lambda \geq 0$ tel qu'il existe un planning réalisable x de meilleure qualité pour les critères de I, avec $\mathsf{card}(E(x)) = \lambda$, si $S_\Pi(I) \neq \emptyset$, sinon retourner "non".

La condition d'optimalité est quelques fois relaxée au profit d'une condition plus faible de faisabilité. C'est-à-dire que si on note $C_\Pi(I)$ l'ensemble des plannings de meilleure qualité pour I, alors la condition d'optimalité :

$$S_{\mathrm{DIM}(\Pi)}(I) = \{\lambda \in \quad \text{tel que : il existe un planning } x \in C_\Pi(I), \text{ vérifiant } \mathsf{card}(E(x)) = \lambda\}$$

est remplacée par la condition de faisabilité :

$$S_{\mathrm{DIM}(\Pi)}(I) = \{\lambda \in \quad \text{tel que : il existe un planning } x \in S_\Pi(I), \text{ vérifiant } \mathsf{card}(E(x)) = \lambda\}.$$

Les sous-problèmes de PEPTP étudiés dans la littérature supposent que toute solution optimale possède le même effectif, faisant de la minimisation de l'effectif un critère à satisfaire. Par conséquent, le problème de dimensionnement se restreint à rechercher ce minimum.
Les problèmes de dimensionnement traités dans la littérature concernent principalement :

- les problèmes de positionnement des repos : la charge est définie sur l'ensemble des jours J (voir PARTOUCHE [1998, Annexe E] pour une synthèse) ;
- les problèmes de construction de grilles : la charge est définie sur l'ensemble des étiquettes EV ou l'ensemble des triplets étiquette-jour-compétence $EV \times J \times S$ (voir NARASIMHAN [2000] pour une synthèse et LIN *et al.* [2000]), ou encore l'ensemble des couples vacation-jour (voir PARTOUCHE [1998, § 8.4]) ;
- Les problèmes de dimensionnement annuel : la charge est estimée sur l'ensemble des couples semaine-compétence (voir WIJNGAARD [1983]).

Ces problèmes sont d'autant plus faciles à résoudre que la charge est cyclique et possède certaines propriétés de régularité. Les tables de besoins sont généralement moins chaotiques que les courbes de charge, d'où l'utilisation de ces problèmes exclusivement dans ce cas.

Comme nous le spécifions en début de section, les classifications sont nécessaires pour élaborer des algorithmes génériques, mais aussi pour concevoir des systèmes informatiques génériques dédiés aux PEPTP. Ce travail de classification pourra être poursuivi de manière à proposer une nomenclature solide d'identification des PEPTP comme il a été fait en ordonnancement[35].

2.3. Bilan Préférentiel et Perspectives

Nous allons ici discuter plus en détails des différentes options empruntées par les concepteurs, pour prendre en compte les préférences des acteurs d'un PEPTP, et pour évaluer des plannings à partir de ces préférences.

2.3.1. Modélisations Usuelles des Préférences et Synthèse Préférentielle

La technique en programmation par contraintes consiste à associer un poids à chaque contrainte mole de propriété. En MIP, cette pratique est équivalente à associer une variable bivalente à chaque contrainte linéaire non obligatoire, qui vaut 1 si et seulement si la contrainte est satisfaite, puis à pondérer cette variable.

Cette pratique est facilement assimilable au scénario introduisant le problème du sac à dos (*knapsack problem*) bien connu en RO. Un randonneur dispose de n objets qu'il envisage d'emporter en randonnée avec lui. Il associe une utilité $c(j)$ à chaque objet j, le poids de j étant $w(j)$. Le campeur cherche à maximiser l'utilité totale de son chargement définie comme la somme des utilités des objets emportés, tout en se soumettant à la contrainte de poids limité

[35] Voir par exemple, BLAZEWICZ J., ECKER K., PESCH E., SCHMIDT G., & WEGLARZ J. [1996] – *Scheduling Computer and Manufacturing Processes* – Springer-Verlag, Berlin, Heidelberg.

supportable par son sac à dos (la somme des poids des objets emportés est limitée). L'utilisation des quotas de points est assimilable à ce scénario : en regardant les points de quota à distribuer comme un pourcentage d'utilité, et la contrainte de poids total limité comme l'ensemble des limites imposées aux plannings. Bien qu'il soit évident pour plusieurs auteurs que ce problème est essentiellement folklorique :

> l'utilisation des poids est assez mal adaptée à la préparation des randonnées car « il ne tient pas compte des utilités "croisées", du fait que le dentifrice sans la brosse à dents et la boite de sardine sans l'ouvre boîtes sont de faible utilité » (SAKAROVITCH [1984*a*]),

Pourtant nous trouvons tout à fait normal d'utiliser la même modélisation pour représenter des préférences individuelles autrement plus complexes en planification. Pour celles-ci non plus il n'est pas rare d'avoir des préférences "croisées".

De plus, aucune place n'est réservée à l'incertitude sur les préférences entre plusieurs situations, etc. Or l'une des caractéristiques des préférences est sûrement l'incertitude, l'information approximative. L'utilisation du nombre est particulièrement mal adapté à la modélisation de cette information.

Depuis la fin des années 60, plusieurs auteurs se sont insurgés contre cette pratique en RO[36]. L'un des gros reproches fait envers ces modèles est l'avis global tranché, clair et précis sur toute comparaison de couple de solutions : un planning x est soit préféré, soit pire, soit équivalent à un autre planning y selon uniquement le résultat de la comparaison de la valeur globale attribuée à x et celle attribuée à y. Aucune place n'est laissée à l'incertitude. Par exemple, deux plannings peuvent être jugés incomparables car ils possèdent chacun des propriétés partiellement incommensurables. Ou encore, la préférence peut ne pas satisfaire la transitivité à cause des approximations faites. Alors que cette incertitude aurait été l'occasion idéale pour les utilisateurs de négocier, discuter sur les compromis possibles.

2.3.2. Préférences et Aide à la Décision : l'Interactivité.

Depuis maintenant quelques années, le concept d'aide à la décision a fait son chemin dans l'élaboration de plannings de travail. Au fil des années, elle a revêtu différentes formes dans le processus aboutissant au choix d'un planning.

Après ce qui vient d'être dit au paragraphe précédent, nous comprenons le succès des techniques basées sur le dialogue avec les utilisateurs. La manipulation (ajout et retrait) de contraintes se font directement par l'utilisateur pour exprimer la subtilité des préférences qui n'a pu être capturé avec des fonctions objectives. L'utilisation d'un arsenal d'indicateurs très intuitifs (compteurs) permettent aux décideurs de disposer d'un meilleur contrôle du processus de décision. Les avancées spectaculaires effectuées en informatique permet maintenant de résoudre des problèmes combinatoires raisonnables en temps réel facilitant ainsi le dialogue avec les utilisateurs.

Mais, les méthodes interactives aussi ont leurs limites. En aide à la décision, il est bien connu que lorsque le nombre de critères et solutions alternatives dépasse un certain nombre[37], le décideur n'est plus aussi efficace. La périodicité de certaines tâches décisionnelles, comme l'élaboration de plannings de travail, permettent au planificateur de devenir au fil des années

---

[36] Ces reproches datent depuis plus de 35 ans avec notamment l'article de Bernard ROY, paru en 1968 au Bulletin de l'AFIRO n7, dont le titre évocateur est : « Il faut désoptimiser la RO ».

[37] Par exemple, Hervé RAYNAUD [2000] fait cette remarque tirée, entre autre, de son expérience en tant que consultant en stratégie industrielle : « Les décideurs ont le sentiment (souvent justifié) de posséder les qualités d'intuition suffisantes pour prendre de bonnes décisions dans le cas où ils ne seraient confrontés qu'à un nombre relativement petits de solutions alternatives et de critères bien définis ». Puis, il ajoute : « Notre expérience tendrait à nous suggérer que l'espèce humaine n'est à l'aise que devant des problèmes dans lesquels le nombre de critères comme celui des solutions alternatives n'excède pas quatre ».

un véritable expert en planification. Ce qui permet par la même occasion de repousser très loin les limites énoncées précédemment. Mais face à un très grand nombre de solutions alternatives, comme c'est le cas des problèmes combinatoires de planification, différents spécialistes de la décision[38] ont constaté un certain nombre de comportements des décideurs dommageables à la décision comme les phénomènes de lassitude et de point d'encrage.

L'auto-planification est une façon de repousser encore plus loin ces limites, mais son champ d'application est pour le moment très restreint. Par contre, elle peut inspirer de nouveaux protocoles de résolution très efficaces.

2.3.3. Complémentarité entre Modélisation des Préférences et Interactivité ?

Qu'est il alors possible de faire ? Nous ne sommes pas ici pour proscrire l'utilisation d'un type de méthodes ou celle d'un autre. Ces méthodes rendent de grands services, et avec les années d'expérimentations sont améliorées. Ainsi, concernant la représentation des préférences et la recherche de compromis, des modèles plus adéquats ont fait leur apparition. Nous nous y attelons aux § 3.3 et chapitre 4. Les progrès effectués en IA concernant les manières d'utiliser les systèmes d'information pour résoudre des problèmes organisationnels, ainsi que la modélisation du raisonnement humain, ouvrent un grand nombre de pistes à explorer pour améliorer l'aide à la décision. Nous nous y consacrons au § 3.2.

Ces deux types de méthodes sont complémentaires. Le reste de ce document est une argumentation pour une ouverture sur ces nouvelles formes de résolution des problèmes de planification, couplant l'interactivité et l'utilisation de modèles préférentiels plus élaborés.

2.3.4. Représentation Implicite des Préférences

Les modèles préférentiels ayant une structure plus faible que celle de la fonction objective ont, pour la plus part, été utilisés pour des problèmes dont l'ensemble des solutions est donné en extension. Par conséquent, une adaptation aux problèmes combinatoires et aux problèmes définis sur des variables continus est nécessaire. Ce sera le thème du chapitre 5, où nous présenterons de nouvelles représentations des relations binaires dans un cadre implicite.

2.3.5. Modélisations de l'Équité.

Nous allons aussi nous intéresser à différentes manières d'exploiter l'information préférentielle. Dans les problèmes d'élaboration de plannings de travail de personnel, les concepts collectifs mis le plus souvent en avant pour traiter les préférences individuelles sont l'unanimité, l'efficacité du système et l'équité. Les deux premières sont très connues, et de plus en plus utilisées dans la littérature, par l'intermédiaire de la dominance de Pareto. L'équité par contre est modélisée de manière très diverse.

Tout d'abord, la quantité d'information jugée intuitivement pertinente pour évaluer l'équité est très volumineuse et complexe. De plus, nous ne savons pas comment la traiter. Par conséquent, les outils usuels sont utilisés : les fonctions statistiques comme la moyenne, la somme simple, etc. ; ou bien la manipulation interactive de contraintes. Prenons l'exemple des plannings.

Dans les problèmes où le personnel est connu (PEPTA et PEPTAG) et où sa participation est primordiale, l'équité est traduite comme :

– l'influence égale en termes de points (ou poids), que chaque employé a dans le processus de décision (WARNER [1976, p. 844], JAUMARD *et al.* [1998, p. 16], OLDENKAMP [1996], RUSHMEIER *et al.* [1995], parmi d'autres). Dans le but d'élaborer un planning, chaque individu reçoit un quota de points qu'il doit répartir entre les différentes propriétés préala-

[38] Voir VINCKE [1989], ROY & BOUYSSOU [1993, § 7.1.3], OTHMANI [1999, § 2.6.2] et leurs références.

blement connues d'un planning (jours de repos, etc.), pour exprimer ses préférences sur les horaires.

- la prise en compte des plannings précédents : le quota de points de chaque individu se voit augmenter en fonction des points non utilisés de plannings précédents qui ont été reportés, et de leur qualité par rapport aux souhaits formulés, etc..

Dans les problèmes impersonnels (PEPTREG et PEPTI), ou dans les PEPTA et PEPTAG non participatifs, l'équité est définie en termes de :

- répartition équilibrée entre les employés, des nombres de matin, après-midi et soirs affectés (JASZKIEWICZ [1997], [2001], PARTOUCHE [1998, § 8.5.1], etc.),
- distribution la moins chaotique possible, de la couverture de la charge (CAI & LI [2000])

Les PEPTPs purement cycliques (§ 2.2.4.3) sont souvent qualifiés de problèmes équitables pour plusieurs raisons :

- chacun a le même planning, à un déphasage près de ses horaires
- les horaires sont connus à l'avance pour au moins une saison. Ce qui permet à chacun de s'organiser à long terme.

Critique : L'utilisation de quotas de points est, nous avons vu, elle-même critiquable (§ 2.3.1). Les plannings cycliques ne tiennent pas compte d'un grand nombre d'information qui nous paraît important pour une évaluation équitable, comme la répartition des jours fériés, la planification des congés payés, etc.. Généralement, il manque un réel modèle permettant de définir clairement ce qui tient d'un jugement équitable et ce qui ne l'est pas.

Solution proposée : Depuis les années 60, plusieurs spécialistes en économie et philosophie, plus généralement en sciences humaines et sociales, se sont intéressés à ce concept pour des problèmes de décision publique (évaluation de la pauvreté, des inégalités, des politiques publiques, etc.). Ils ont ainsi proposé différents modèles qui ont surmonté jusqu'à maintenant l'essentiel des critiques théoriques et pratiques. Nous nous sommes intéressés à ces modèles. Ceux-ci s'insèrent naturellement dans le type de problèmes que nous étudions. De plus, elles sont entièrement en accord avec les nouveaux modèles représentant les préférences et avec l'utilisation de l'interaction entre l'homme et le système d'information. Cet aspect est largement débattu au chapitre 6.

3. Avancées en Modélisation : Aide à la Décision et Préférences Individuelles

« If a concept has some basic ambiguity, then a *precise* representation of that ambiguous concept must *preserve* that ambiguity, rather than try to remove it »
extrait de SEN [1992 § 3.4] et FOSTER & SEN [1997, p.121].

Face à la variété des approches possibles du problème d'élaboration de planning de personnel avec un ordinateur, on peut se poser la question de la place que peut occuper l'ordinateur dans la résolution concrète de problèmes en contexte organisationnel. Comment prendre une meilleure décision pour les protagonistes en usant de tout le potentiel humain et artificiel (ordinateurs) à notre disposition ? Comment s'affranchir au mieux des limites engendrées par la résolution exclusivement humaine ? La résolution entièrement automatique n'est pas toujours acceptable. Comment s'affranchir au mieux des limites engendrées par la résolution exclusivement automatique ? Les réponses à ces questions permettront de concevoir des outils logiciels plus adéquats, voire plus ergonomiques dédiés à la résolution de notre problème. Ce type de questions est central à l'intelligence artificielle dont l'un des buts est justement d'étudier comment aborder les problèmes complexes au moyen d'un ordinateur. Nous en discutons au § 3.2.

Un autre problème est aussi apparu en étudiant les modèles existants. C'est celui de la représentation des préférences. Comment peut-on modéliser les préférences de manière à obtenir des modèles adaptés au traitement sur ordinateur et à la communication avec l'être humain ? Comment peut-on modéliser des préférences dans un contexte implicite ? Comment traiter cette information pour faire progresser la résolution d'un problème ? Plusieurs disciplines se sont intéressées à la compréhension et représentation des préférences. Nous les étudions dans ce chapitre et au suivant. Ainsi, le présent chapitre, plus particulièrement le § 3.3, fait un état de l'art des différents modèles existants pour représenter les préférences d'un individu, et étudie comment cette information préférentielle peut être utilisée pour apporter des réponses à un problème de choix (§ 3.4). Tandis que le chapitre suivant s'intéresse à la modélisation (§ 4.1), et à la prise de décision (§ 4.2) dans un contexte multicritère et multi-acteurs. Cette étude nous conduira à proposer :

- de nouveaux axiomes pour le choix de meilleures solutions lorsque l'optimalité et la maximalité n'a plus de sens (§ 3.4),
- de baser le traitement de l'information préférentielle sur l'analyse axiomatique (§ 4.2.2),
- un modèle pour la représentation implicite d'information préférentielle ordinale (§ 5.2), et enfin,
- de nouvelles problématiques d'aide à la décision pour améliorer la résolution de problèmes partiellement automatisés (§ 5.3),

3.1. Préliminaires : Aide Multicritère à la Décision et Choix Social

Les deux domaines de recherche auxquels s'apparente notre quête d'un meilleur modèle véhiculant l'information préférentielle, sont l'aide multicritère à la décision et la théorie du choix social. Ces deux disciplines, distinctes par leur origine, le vocabulaire utilisé et les ap-

proches préconisées, ont fait des préférences leur cheval de bataille, pour résoudre leurs problèmes. Différentes disciplines se sont aussi intéressées à la place de l'ordinateur dans une résolution de problèmes de décision : la branche traitement de l'information de l'intelligence artificielle et l'aide multicritère à la décision. Nous présentons ces trois disciplines ci-dessous.

3.1.1. Qu'est-ce que l'Aide Multicritère à la Décision ?

L'**aide multicritère à la décision** (AMCD)[39] s'est développée depuis la fin des années 60 pour faire face aux lacunes de la méthodologie monocritère classique de la recherche opérationnelle. Cette méthodologie conçoit le modèle d'un problème, avec un vrai-critère unique, pour qu'il soit résolu de manière entièrement automatique par des algorithmes d'optimisation simple. Hélas, ces modèles ne reflètent pas toujours la réalité des problèmes.

L'étude du comportement humain face à un problème, la manière dont les décisions sont prises habituellement, ont fait apparaître le caractère multiple souvent contradictoire et incommensurable de l'information préférentielle. Une évolution s'est donc produite vers la prise de décision en présence de critères multiples souvent conflictuels[40].

Les problèmes pratiques rencontrés ont conduit à élaborer un grand nombre de méthodes plus ou moins pragmatiques.

Les problèmes de la RO encourageant vivement une aide multicritère à la décision sont nombreux. Ils vont des problèmes d'ordonnancement de tâches, aux problèmes de gestion de personnel, en passant par les problèmes de réseaux d'affectation et de tournées de distribution. L'aide multicritère à la décision s'est aussi proposée comme moyen d'analyse et de préparation d'un grand nombre de décisions dans les organisations : problèmes de mercatique, de gestion financière, de sélection et évaluation de projets, etc.

Par contre, les techniques développées sont moins bien adaptées à un certain nombre de problèmes plus stratégiques (ROY & BOUYSSOU [1993]), politiques ou encore éthiques[41] (doit-on vendre des armes à tel pays ? Qui doit-on faire bénéficier en priorité des transplantations d'organes ? Doit-on aider tel pays à développer son industrie nucléaire ? Doit-on accorder l'indépendance à certains territoires ?[42]).

3.1.2. Qu'est-ce que la Théorie du Choix Social ?

La **théorie du choix social** (ou théorie du choix collectif ou théorie de la décision sociale, TCS en abrégé)[43] est concernée principalement par les relations existant entre les préférences des individus d'une société et les choix effectués au niveau collectif. Apparue en tant que discipline à part entière des sciences humaines et sociales dans les années 50 (avec les travaux d'ARROW [1951]), ses origines remontent au XVIIIe siècle[44]. La TCS s'intéresse à différents types de problèmes de décision publique comme[45] :

– Le choix de procédures pour les décisions de comités,

---

[39] Pour des ouvrages de référence, voir ROY [1985], ROY & BOUYSSOU [1993], VINCKE [1989].

[40] Les fortes relations qu'il a été donné à quelques spécialistes de la RO d'avoir avec des milieux non académiques et avec des spécialistes de diverses disciplines, ont favorisé l'émergence de ce nouveau paradigme. ROY [1969-1970] et KEENEY & RAÏFFA [1976] constituent deux ouvrages historiques dévoilant les deux principales facettes de l'aide multicritère à la décision telle qu'elle se présente aujourd'hui.

[41] L'éthique est la partie de la philosophie qui étudie les fondements de la morale.

[42] Exemples tirés de MOUSSEAU [1993].

[43] Pour un article de référence, voir SEN [1986]. Voir aussi SEN [1970], ARROW *et al.* [1997], [2002]. Pour une liste d'articles fondamentaux parus sur le sujet, voir ROWLEY [1993].

[44] SEN [1986] retrace les origines de cette théorie qui se sépare en deux sources distinctes. La première est la théorie mathématique des élections et des décisions de comités, avec les travaux de BORDA (1781) et CONDORCET (1785). La seconde source est l'utilitarisme et la théorie du bien-être, amorcés par les travaux de BENTHAM (1789) entre autre.

[45] Exemples tirés de SEN [1986, p. 1160]. Voir aussi SEN [1999] et [1977].

- L'examen de règles électorales,
- Le choix d'une constitution pour un nouveau pays indépendant,
- Le jugement sur le fait que le gouvernement d'un pays ait failli dans sa fonction de serviteur des intérêts de la nation,
- Le choix de méthodes d'évaluation de politiques fiscales,
- La réalisation de plannings centraux basés sur les intérêts de la communauté,
- La formulation systématique de jugements de bien-être social,
- L'exploration de concepts moraux (tels que l'équité, la justice, la liberté, l'égalité, les droits),
- La construction d'indicateurs significatifs d'un point de vue éthique de la prospérité, de la pauvreté ou de l'inégalité nationales,

Ces problèmes ont en commun de vouloir émettre des jugements collectifs entre les différents états sociaux probables (ou solutions) sur la base d'informations préférentielles formulées par différents individus. Les différences sont multiples. Elles vont de l'importance accordée aux considérations éthiques et pragmatiques, à la nature des préférences individuelles et collectives (information dont dispose chaque acteur, confiance accordée aux préférences formulées, contraintes structurales imposées, etc.).

Souvent présentée dans le cadre des sciences politiques – certainement de par son passé[46] – , la TCS entretien aujourd'hui des liens très étroits avec d'autres disciplines, notamment la théorie des jeux, la philosophie morale et sociale, etc.. Nous renvoyons à BARNETT *et al.* [1995] pour un surplus d'information.

Discipline analytique, la TCS fait une utilisation étendue, presqu'exclusive des méthodes axiomatiques. Ses forces et ses faiblesses viennent précisément de ce caractère analytique.
Les méthodes axiomatiques consistent, très brièvement, à énoncer un ensemble d'axiomes de description d'un problème, et de principes jugés pertinents pour ce problème, puis d'étudier leur cohérence pris conjointement.
Ainsi, les écrits sur la TCS abondent en résultats d'incohérence (théorèmes d'impossibilité) et de cohérence (théorèmes de caractérisation constructive) de systèmes axiomatiques.

3.1.3. Intelligence Artificielle et Traitement de l'Information

La branche traitement de l'information (ou computo-symbolique) de l'IA (notée IACS) désigne l'ensemble des techniques utilisées en informatique pour programmer des processus complexes[47]. Plus précisément, elle a pour objectif de développer des outils capables de manipuler des concepts et d'effectuer des actions sur des objets symboliques[48] « qui, il y a un siècle, auraient été considérés comme une faculté humaine uniquement ». Apparue dans les années 50, le traitement de l'information s'intéresse entre autres à[49] développer la lisibilité des processus qu'elle modélise : Les systèmes conçus mettent en avant, non pas le fait qu'ils modélisent ou non une intelligence humaine, mais plutôt le fait qu'une intelligence humaine puisse comprendre son comportement en termes de concepts clairement définis.

3.1.4. Une Présentation Commune Générique

---

[46] Revoir note en bas de page numéro 44.

[47] SIMON (1957) fut l'un des pères fondateurs de l'intelligence artificielle. Il s'appuie sur le paradigme comportemental suivant : « *Pour résoudre des problèmes avec des machines, faisons comme les gens font* ! ». Les travaux de Herbert A. SIMON, économiste de formation, ont aussi influencé la TCS (voir par exemple SEN [1987]). Mais son attirance a été plus probante en AMCD.

[48] Par opposition aux chiffres et au calcul numérique.

[49] Pour plus de détails, voir KODRATOFF Y. [1989] – IA et approche connexionniste – *Bulletin de liaison de la recherche en informatique et en automatique de l'INRIA* 124 : 16-21. Voir aussi GARBAY [2003].

La TCS et l'AMCD ont plusieurs points communs, à commencer par le principe qui consiste à :

« **Établir une information préférentielle globale ou collective exploitable, sur la base d'un ensemble de critères ou de préférences individuelles**. »

Les divergences sont aussi nombreuses, dues vraisemblablement aux problèmes traités et aux approches préconisées, qui sont différentes. Nous ne ferons pas ici de comparaison, et dirigeons le lecteur vers les références ROY & BOUYSSOU [1993, § 3.2.4], DURAND [2000, § 1.1.2].

Nos problèmes possèdent plusieurs particularités qui, pour une modélisation et résolution de qualité, nécessitent l'utilisation conjointe de ces trois disciplines. L'AMCD dispose d'une expérience pratique et algorithmique importante. Elle propose ainsi un cadre relativement riche pour la conception de logiciels dédiés aux problèmes multicritères. La TCS, elle, apporte son expérience analytique, ainsi qu'une dimension sociale et éthique du comportement humain (aux préférences). L'IACS offre des outils de gestion des processus décisionnels allant de l'interactivité (ergonomie) à la formulation et résolution de problèmes partiels pertinents.

3.2. Modèles, Processus de Décision et Aide à la Décision

Pour concevoir un système informatique adéquat pour une prise de décision dans une organisation, il est auparavant nécessaire d'avoir une réflexion sur la place que peut occuper un ordinateur dans la compréhension et la résolution d'un problème posé. Comment voyons-nous la résolution d'un problème réel par l'intermédiaire d'un ordinateur ? Quel cheminement l'homme effectue-t-il pour concevoir un modèle ? Quelle est la place du dialogue et du calcul dans un processus de décision ? Attardons-nous dans cette section sur ces questions.

3.2.1. Conception de Modèles et Aide à la Décision

3.2.1.1. Système de Traitement de l'Information et Aide à la Décision

Les décisions en entreprise peuvent se classer en plusieurs groupes. Certaines sont faciles à prendre et se font par ajustement mutuel (MINTZBERG [1998]). D'autres obéissent à un programme dont le déroulement est immuablement fixe. Dans ce cadre, certaines décisions intègrent un aspect combinatoire qui peut être traité entièrement par un ordinateur.

Mais la plupart des tâches dans une organisation combinent les deux aspects : le dialogue et la combinatoire. Pour ce type de tâches, la machine devient véritablement un acteur supplémentaire dans la prise de décision. Malgré cela, elle ne doit pas être perçue par les acteurs humains comme une contrainte, ou un outil imposant ses propres décisions programmées par le concepteur sans laisser de marge aux vrais acteurs. Par exemple, un système qui statue sur des conflits sans accord préalable des concernés, alors que la résultante de ces conflits est fortement dépendant des acteurs et de l'environnement du problème. Dans une conférence de l'AFCET en 1993, retranscrite sous forme textuelle dans COURBON *et al*. [1994], Didier DUBOIS disait que : « *Quand vous poser une question à un système d'information, vous n'attendez pas qu'il choisisse pour vous. Vous attendez qu'il fasse une déduction, raisonnement plausible, mais pas que le système d'information prenne une décision.* ». Le système d'information doit être perçu comme un outil d'**aide à la décision**. Le contrôle du processus de décision par les utilisateurs devrait occuper une place prépondérante.

Entre l'homme et la machine, chacun a sa spécialité. L'ordinateur est plus doué pour un certain nombre de tâches que l'être humain : le traitement (la manipulation) de données, le calcul combinatoire, la mémorisation non biaisée d'information (LÉVINE & POMEROL [1989,

§ 11.2]), etc.. L'ordinateur est en d'autres termes un **système de traitement de l'information** et de stockage fiable des données.

D'un autre coté, concevoir un système informatique dédié à un type de tâche dans une organisation nécessite une forme élaborée de communication entre l'homme et le système. Le dialogue d'humain à humain est autrement plus facile qu'un dialogue d'humain à machine. La machine nécessite de représenter par un modèle l'information, ce qui n'est pas toujours chose aisée[50], malgré les avancées en IHM, représentation des connaissances (et plus généralement en intelligence artificielle) repoussant sans cesse les limites. Communiquer qualitativement avec un être humain est très difficile.

3.2.1.2. La Conception de Modèles

L'ordinateur a besoin d'une représentation, nous disons un modèle, pour comprendre et traiter un problème. Deux parties se dégagent dans la définition d'un modèle d'un système réel :

- Le modèle de faisabilité pour une description en compréhension ou en extension des solutions ;
- Le modèle préférentiel pour l'évaluation de ces solutions.

Pour construire un modèle d'un problème réel, la technique universelle est de diviser pour régner. Nous nous appuyons sur des problèmes partiels ou des sous-problèmes que nous arrivons à résoudre. L'exhaustivité est alors de mise. Puis, nous essayons de recoller les morceaux pour faire un problème à peu près défini.

Lorsqu'un problème du monde réel est modélisé, le type d'instance est décrit (de quoi il est formé, quels sont les attributs pertinents) aboutissant ainsi à la connaissance de $\mathcal{I}$, et le type de solution est aussi décrit (de quoi est formé une solution) aboutissant à la connaissance de $\mathcal{S}$ Puis, la relation $\Pi \subseteq \mathcal{I} \times \mathcal{S}$ est décrite au moins partiellement. Toutes ces informations constituent un modèle du problème réel.

Quelles sont les questions auxquelles le concepteur répond lorsqu'il modélise un problème réel ? La description de Π (comment est décrit le lien entre les instances et les solutions) peut correspondre au problème d'appartenance associé à Π : Étant donné un couple $(I, x) \in \mathcal{I} \times \mathcal{S}$, est ce que x est une solution réalisable pour I ($\Leftrightarrow x \in S_\Pi(I)$) ? Cette description n'implique pas forcément que l'on pourra toujours trouver par voie algorithmique une solution réalisable, et lorsque c'est possible, de la trouver en un temps polynomial[51]. Chacun des problèmes partiels de Π intuitivement solvables est un axiome. Par axiome, nous entendons 'hypothèse de travail', dans le même ordre de pensée que PERNY [1992] (voir aussi Équipe MUSTIL [1996, section 2]).

Les problèmes combinatoires et continus imposent une difficulté supplémentaire, à savoir que pour une instance donnée, l'ensemble $S_\Pi(I)$ n'est pas en extension. L'une des techniques de modélisation qui a fait ses preuves, est de décrire le problème par ses contraintes, la recomposition se faisant généralement par une conjonction formant un système de contraintes. Par contre, ce n'est pas la seule description possible. Par exemple, les problèmes dynamiques (comme les processus stochastiques) sont généralement décrit en introduisant des paramètres d'évolution (le temps par exemple). La description intègre des propriétés se conservant au cours du temps. Lorsque les solutions ne sont pas connus explicitement, bien d'autres problèmes partiels de Π – et donc plus faciles que Π en terme de complexité (§ 2.1.1.2) – peuvent être résolus, et forment le modèle de départ.

---

[50] LÉVINE & POMEROL [1989 p. 19] notent d'ailleurs qu'« une représentation [de l'information] est succeptible d'être traitée comme un nouveau problème à résoudre ».

[51] Voir les problèmes indécidables en théorie de la complexité (GAREY & JOHNSON [1979], PAPADIMITRIOU [1994])

Vient aussi la description du modèle préférentiel, lequel s'effectue, en questionnant les différents protagonistes sur les couples (I, x) qui correspondrait le mieux à une bonne solution pour eux. Les avis ne convergeant qu'exceptionnellement, il est nécessaire que chacun explique au mieux ses préférences sur l'ensemble des solutions. En fonction du temps imparti, l'accessibilité des collecteurs d'information (être humain ou machine), cette information préférentielle peut être plus ou moins claire. La description effectuée par chaque personne répond généralement aux problèmes partiels suivants :

problème de comparaison : Étant donné un couple $(x, y) \in \mathcal{S} \times \mathcal{S}$, quel jugement préférentiel établissez-vous entre x et y ?

La réponse est très variable, voire dépendante de l'interlocuteur. Dans le cas classique, la réponse est : mieux, pire ou indifférents.

problème d'évaluation : Étant donné $x \in \mathcal{S}$, quelle est la <u>valeur</u> attribuée à x ?

Le terme « valeur » désigne n'importe quel attribut modélisé par un nombre, ou une quantité (comme le « coût », le « profit », l'« utilité » en économie, etc.), voire par une échelle qualitative (par exemple : {excellent, bien, moyen}).
Lorsque le problème de comparaison est basé sur les réponses {mieux, pire, indifférents}, ces deux problèmes sont liés par la relation suivante : Dire que x est mieux que y est équivalent à dire que la valeur de x est supérieure (ou inférieure, selon l'objectif), à celle de y.
D'autres problèmes de base peuvent être utilisés, comme le **problème de vérification de l'optimalité** (*check optimality problem*) associé à un problème d'optimisation simple : Est-ce qu'une solution donnée est optimale pour une instance de problème donnée ? Remarquons que ce problème est rarement intuitif, voire souvent très difficile. Assez souvent, il est difficile pour un acteur de répondre aux questions de comparaison ou d'évaluation. Pour y remédier, un acteur peut avoir plusieurs critères d'évaluation d'une solution et de comparaison d'un couple de solutions (ROY [1985]).
Cette description n'implique pas toujours qu'une solution de meilleure qualité existe, ni que l'on pourra trouver une telle solution par voie algorithmique. Les problèmes combinatoires et continus imposent une difficulté supplémentaire, à savoir qu'une telle description, couple de solutions par couple de solution, n'est pas physiquement réalisable à cause du nombre astronomique de solutions, voire l'indénombrabilité (dans la réalité) de l'ensemble $S_\Pi(I)$. Les fonctions objectives ont permis d'y remédier. Nous présentons d'autres possibilités au chapitre 5.

3.2.1.3. Critères de Modélisation

Un modèle dépend des critères de conception valorisés.

L'exhaustivité des problèmes partiels de Π qui sont intuitivement solvables (auxquels nous pouvons répondre sans effort cognitif notable), est souvent recherchée. Mais elle est très difficile à obtenir et impensable pour des problèmes réels. En effet, il en existe une nébuleuse. Le mieux est d'être exhaustif en évitant les redondances d'information, ce qui n'est pas toujours possible :

Un autre problème s'ajoute lorsque nous modélisons : Certains problèmes partiels nous paraissent faciles à résoudre à cause de l'approche que nous avons du problème. Par exemple, la résolution directe du problème de comparaison associé à Π sur l'ensemble des couples de solutions de Π n'est pas humainement énumérable pour les problèmes admettant une représentation implicite des solutions. Il est alors nécessaire de considérer des problèmes de complexité plus élevée. Par exemple : Toute solution x vérifiant un certain nombre de propriétés est préféré à une solution y vérifiant un autre ensemble de propriétés. De tels axiomes, bien

qu'humainement énumérables augmentent le pourcentage de cas particuliers où certains axiomes sont difficilement justifiables ou sont incompatibles entre eux.

Les axiomes doivent être perçus donc comme des hypothèses de travail, et non comme des vérités indubitables (voir PERNY [1992, p. 28]). Sur de telles bases, on comprend que les modèles d'optimisation simple n'ont eux-mêmes rien à se reprocher. C'est plutôt, le trop grand écart qui existe entre le modèle et le système réel qui pose problème. Dans le cadre de la planification, tout un travail de synthèse, traitement de l'information fourni par les autres acteurs, sans aide aucune, doit être fait par le planificateur pour se ramener à un problème bien posé. Ce travail est généralement très difficile. Dire qu'un problème d'optimisation simple modélise un PEPTP, c'est dire que le statut d'hypothèse de travail des axiomes formant le modèle est négligé.

En définitive, quatre critères de modélisation sont identifiés : l'exhaustivité, la non redondance, l'énumération humainement réalisable, la possibilité de remettre en question certains axiomes (hypothèses de travail) lors du processus de décision.

3.2.2. Systèmes Interactifs d'Aide à la Décision et Processus de Décision

Les systèmes informatiques peuvent être classés selon l'importance qu'elles accordent aux interventions du décideur en cours de traitement et selon le rapport traitements/données. La Figure 10 présente les 4 principales classes existantes.

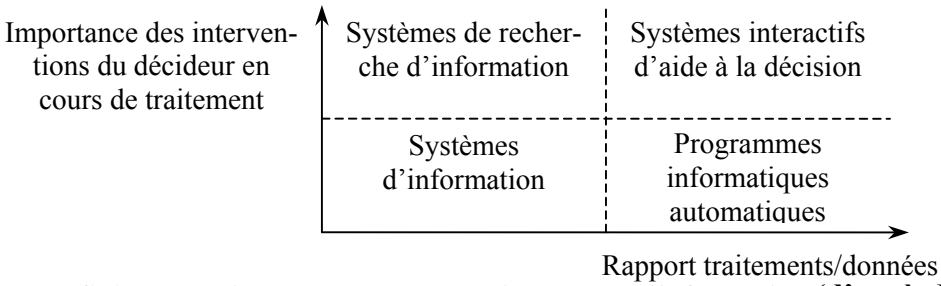


Figure 10 : Les SIADs au sein des systèmes de traitement de l'information (d'après LÉVINE & POMEROL [1989]).

Les PEPTP ont souvent été conçu comme des programmes informatiques automatiques, à cause de leur aspect combinatoire, l'aspect négociation étant négligé. L'importance de la négociation a poussé beaucoup de concepteurs à reconsidérer ce choix, et à les classer parmi les systèmes interactifs d'aide à la décision. Nous les présentons ci-dessous :

3.2.2.1. Systèmes Interactifs d'Aide à la Décision

Un **système interactif d'aide à la décision** (SIAD, *decision support system*) est un système de traitement de l'information, destiné à résoudre des problèmes décisionnels, en laissant de l'initiative à son utilisateur.

LÉVINE & POMEROL [1989, p. 13] remarquent que chaque lettre formant SIAD est une clé pour l'analyse du concept dans son ensemble. Nous reprenons leur propos : Le mot « système » réfère à une architecture complexe que l'on peut analyser comme un tout, le mot « interactif » renvoie au rôle indispensable de l'homme dans le fonctionnement du SIAD, rôle qui n'est pas passif comme l'indique « aide à la décision », ce qui sous-entend que la décision ne sera pas prise par le système.

Tout l'art de la conception d'un SIAD réside dans l'instauration d'un véritable dialogue entre l'homme et la machine. Pour que cet échange soit fructueux, il est nécessaire d'identifier les problèmes partiels automatisables et pertinents pour l'utilisateur, pouvant faire avancer la résolution.

Décrivons de manière plus technique l'environnement d'un SIAD en utilisant la notation de la théorie de la complexité (§ 2.1.1.2). Dans le cadre général, un **problème réel partiellement automatisé**[52] possède les caractéristiques suivantes :

- Une description de l'ensemble des instances $\mathcal{I}_{\Pi 0}$ et de l'ensemble des solutions $\mathcal{S}_{\Pi 0}$ est connue ;
- Aucune relation $\Pi 0 \subseteq \mathcal{I}_{\Pi 0} \times \mathcal{S}_{\Pi 0}$ n'a pu être clairement identifiée pour le décrire. Deux cas se présentent alors pour une instance $I_{\Pi 0}$: Soit le modèle actuel ne permet pas d'identifier les solutions réalisables de cette instance ($\Leftrightarrow S(I_{\Pi 0})$ est inconnu), soit les préférences $pref(I_{\Pi 0})$ qui ont pu être modélisées dans $I_{\Pi 0}$ ne permettent pas d'identifier une solution de meilleure qualité ;
- Par contre, une étude préalable a permis d'identifier un certain nombre de problèmes partiels entièrement automatisables $\Pi 1, \Pi 2, \ldots, \Pi p$, appelés **problèmes d'aide à la décision** ou **problèmes techniques**, et pour chacun desquels un algorithme de résolution respectivement $A_{\Pi 1}, A_{\Pi 2}, \ldots, A_{\Pi p}$ est connu. Un problème partiel d'un problème Π donné correspond à un problème dont les réponses peuvent faire progresser la recherche d'une solution du problème initial Π.

Parmi les problèmes d'aide à la décision, il peut y avoir des sous-problèmes de $\Pi 0$. De plus, la problématique décisionnelle de $\Pi 0$ peut être entièrement différente de celle des problèmes d'aide à la décision.

`DecisionProcess`($I_{\Pi 0} \in \mathcal{I}_{\Pi 0}$) : **retourner** Structure de Stockage
début

1) $x \in \mathcal{S}_{\Pi 1} \cup \mathcal{S}_{\Pi 2} \cup \ldots \cup \mathcal{S}_{\Pi p} \cup \{\text{non}\} \leftarrow$ non — Solution courante, valant "non" lorsqu'il n'en existe pas.
2) estSatisfaisante : Booléen ← non — Booléen qui vaut "oui" si le décideur est satisfait de la proposition faite, et de la situation courante.
3) DB : Structure de Stockage ← $I_{\Pi 0}$ — Information stockée en cours de processus.
4) **tant que** (estSatisfaisante = non) **faire**

Étape de dialogue

5) présenter information (DB)
6) estSatisfaisante ← la proposition est-elle satisfaisante ?
7) **si** (estSatisfaisante = oui) **alors** **retourner** DB **fin si**
   DB contient la solution choisie, mais aussi les justificatifs du choix effectué.
8) $DB \leftarrow$ compléments d'information(DB, $\{\Pi 1, \Pi 2, \ldots, \Pi p\}$)
9) $(\Pi, I_{\Pi}) \leftarrow$ choisir problème à résoudre(DB, $\{\Pi 1, \Pi 2, \ldots, \Pi p\}$)
   Identification d'un nouveau problème partiel pouvant faire progresser la recherche. Construction de l'instance à résoudre à partir de DB et autre information recueillie par interaction.

Étape de calcul

10) $x \leftarrow A_{\Pi}(I_{\Pi})$
11) $DB \leftarrow$ Stocker(DB, I_{Π}, x) — Stockage de l'information pertinente en tenant aussi compte des limites de la mémoire.

fin tant que
fin `DecisionProcess`

Algorithme 1 : Processus de décision général lorsqu'il est effectué par un décideur unique.

Les SIADs fournissent un cadre général pour résoudre les problèmes partiellement automatisés, c'est-à-dire, trouver une solution **satisfaisante** à l'instance $I_{\Pi 0}$ du problème initial

[52] Les problèmes réels partiellement automatisés sont aussi appelés problèmes partiellement normalisés (LÉVINE & POMEROL [1989]), ou problèmes mal-posés.

$\Pi 0$. Dans le cas particulier où la résolution se déroule avec un unique décideur, alors le processus de décision est une succession d'étapes de dialogue avec le décideur et de calcul (Algorithme 1). Dans ce cadre, les trois éléments caractéristiques des SIADs sont :

- Une structure de stockage de l'information, ou base de données *DB*,
- Des problèmes d'aide à la décision $\Pi 1$, $\Pi 2$, …, Πp, et les algorithmes de résolution associés $A_{\Pi 1}$, $A_{\Pi 2}$, …, $A_{\Pi p}$, et
- Une interface qui assure un dialogue avec l'utilisateur tout au long de la résolution.

Ce schéma très général témoigne de la variété des approches possibles et légitimes de ces problèmes, et illustre bien la richesse incontestable de ce domaine.

Quelques propriétés d'un processus de décision :

- Le processus retourne la structure *BD* décrivant la manière dont s'est déroulé le processus de décision, c'est-à-dire les différentes étapes et décisions partielles par lesquelles l'utilisateur est passé. Ainsi, *BD* contient la ou les solutions retenues, ainsi que les justificatifs de cette décision globale, de manière à rendre compte auprès des autres acteurs concernés (voir § 1.2.4).
- Un certain contrôle des problèmes accessibles à une itération donnée, est nécessaire pour une conduite du processus, voire une convergence de la résolution.
- L'information préférentielle $pref(I_{\Pi 0})$ de $I_{\Pi 0}$ sert, entre autre, à appuyer et justifier les décisions prises à chaque étape d'interfaçage avec le décideur.
- La situation (état final) est satisfaisante (estSatisfaisante = oui) pour le décideur dans le processus, lorsque celui-ci est effectivement satisfaisant, mais aussi vu le temps qui lui est imparti, la patience dont il fait preuve (OTHMANI [1998), le rôle et le pouvoir qui lui est accordé pour effectuer cette tâche de recherche d'une solution.
- Ce processus est souvent qualifié d' « individuel », car la dynamique créée par le processus ne concerne qu'un individu (ROY & BOUYSSOU [1993, § 7.1.3]). La prise de décision est individuelle.

Voici quelques concepts clé des SIADs :

3.2.2.2. Recherche Heuristique

Un aspect qui n'est pas du tout mis en avant par le processus général présenté par l'Algorithme 1 est la recherche heuristique. Elle décrit la manière dont l'utilisateur utilise le processus de décision : Certaines décisions partielles sont révocables. En d'autres termes, la recherche de solutions est assimilable au parcours d'une arborescence. Les sommets correspondent aux différentes situations possibles, le passage d'une situation à une autre s'identifiant aux décisions locales effectuées par l'utilisateur lors d'une étape de dialogue. Certaines branches de l'arborescence se soldent par un échec, commandant alors un retour arrière vers un état précédent (NEWELL & SIMON [1972], SIMON [1977], LÉVINE & POMEROL [1989]).

3.2.2.3. Gestion des Données

La structure de stockage *DB*, ou mémoire, est initialisé à $I_{\Pi 0}$, ce dernier pouvant ne contenir aucune information. En cours de processus, elle stocke l'information intermédiaire. Par exemple le couple (instance, solution) obtenu après chaque étape de calcul.

Un stockage de l'information pertinente doit tenir compte des limites de la mémoire. Les programmes tirent partie des dépendances entre problèmes stockés (on parle alors d'apprentissage : § 3.2.2.6). L'utilisateur peut, lui aussi, imposer ses choix de stockage au cours du processus de décision.

La nécessité de stocker une grande masse d'information, doit être gérée intelligemment. D'où la nécessité d'un **système de gestion de base de données**. On parle aussi de gestion optimisée des données.

Nous ne nous attardons pas sur ce point, et renvoyons à LÉVINE & POMEROL [1989].

3.2.2.4. Interactivité

Les étapes de saisie et de restitution d'information (dialogue) sont essentielles au bon déroulement d'un processus de décision. Il est nécessaire de rechercher des représentations pour certains concepts clés utilisant les différentes dimensions qu'un ordinateur met à disposition (écran, couleurs, etc.). On parle alors d'interface homme-machine (IHM) et d'ergonomie.

Il peut exister plusieurs représentations d'un même objet, montrant ses différentes facettes de manière claire et concise. Par exemple, différentes formes pour les plannings ont fait leur apparition (voir PARTOUCHE [1998], CHAN [2002]).

3.2.2.5. Filtre Intelligent de l'Information

C'est un autre aspect qui ne transparaît pas dans la description (Algorithme 1) du processus de décision général. Face à l'accroissement extraordinaire des informations, du savoir et de la complexité lié à l'intrication des phénomènes, un SIAD doit être pour son utilisateur un **filtre** : Il ne doit laisser passer qu'une quantité d'information analysable par un opérateur humain, et de préférence les bonnes informations (SIMON [1977], LÉVINE & POMEROL [1989, § 11.2 et 11.3]). Cette pertinence peut aller jusqu'à contredire le décideur dans ces jugements, sachant que l'esprit humain peut être soumis à des biais cognitifs. L'objectif prioritaire du SIAD est d'élargir la perspective du décideur, plutôt que de l'entretenir dans son conformisme (rôle prescriptif ou actif).

3.2.2.6. Gestion des Problèmes d'Aide à la Décision : Apprentissage

Un autre aspect important des processus non visible à partir de l'Algorithme 1 est la réutilisation de l'information intermédiaire générée par un problème d'aide à la décision. L'Algorithme 1 stocke en mémoire uniquement les instances et les solutions des différents problèmes partiels résolus de manière séquentielle. L'information intermédiaire produite lors de la résolution d'un problème partiel peut s'avérer pertinente pour des étapes ultérieures. Par conséquent, pour ne pas avoir à recalculer cette information intermédiaire pertinente, une possibilité est de la stocker. Un processus d'aide à la décision est ponctué de phases de stockage d'information intermédiaire. Nous ne reviendrons pas ici sur la nécessité d'établir un compromis entre re-calcul et stockage limité.

Tout problème peut se décomposer en problèmes partiels, qui à leur tour se décomposent en problèmes partiels, ceci de façon récursive jusqu'à ce qu'on arrive aux instructions élémentaires, qui elles sont indivisibles. Chaque instruction élémentaire peut être considérée comme un problème partiel du processus de décision. Un problème d'aide à la décision est un problème partiel qui se situe entre deux interventions de l'utilisateur. Ce problème est formé d'une séquence de problèmes partiels pertinents pour une réutilisation. Chaque solution et instance de problème partiel pertinent sont stockés en mémoire. Ce stockage d'information non contrôlée par l'utilisateur permet souvent d'effectuer un gain considérable dans le traitement de l'information. Une certaine forme d'apprentissage est alors réalisée par le système. LÉVINE & POMEROL [1989] parlent alors de **système de gestion de base de modèles**, le terme modèle étant employé pour désigner le couple (problème d'aide à la décision, algorithme de résolution). Ces dépendances ont été exploitées notamment dans les problèmes d'explication de l'inconsistance par VERFAILLIE & LOBJOIS [1999] (§ 3.2.3.5).

Les systèmes de gestion des données et des modèles, formant l'architecture du SIAD, peuvent être intimement liés. LÉVINE & POMEROL [1989] décrivent différents types d'architecture au chapitre 4 de leur livre.

3.2.2.7. Systèmes Interactifs d'Aide à la Décision en Groupe

Lorsque plusieurs acteurs interviennent dans le déroulement du processus de décision, deux cas différents se présentent :

- Les négociations s'effectuant en cours de processus sont gérées par les acteurs eux-mêmes, sans que rien ne soit changé dans le processus de décision individuel (Algorithme 1). Les acteurs agissent envers le système informatique comme un seul décideur, les négociations étant extérieures au (indépendantes du) système.
- Le système est conçu de manière à faciliter le travail et la décision de groupe. L'interaction mono-décideur et l'aspect base de données sont remplacés par un module « fusionnel » de première importance, basé sur les technologies de « groupware » : suivi et contrôle de qualité du processus de décision, accès et mise en forme de l'information (COURBON *et al*. [1994 p. 23]).

Ces systèmes subissent un développement croissant depuis le début des années 90, et sont appelés GDSS (*group decision support systems*) et NSS (*negotiation support systems*). Nous n'approfondirons pas plus, et renvoyons par exemple aux références de VINCKE [1989 § 7.10][53].

3.2.3. Problématiques d'Aide à la Décision

Un grand nombre des caractéristiques d'un SIAD décrites précédemment sont mises en œuvre dans les problèmes d'aide à la décision. Certains nécessitent un traitement lourd, d'autres non. Comme nous l'avons signalé au § 2.1.1, les problèmes sont caractérisés par leur diversité. Face à cette variété, il est nécessaire de les classer. L'AMCD (ROY [1985], BANA E COSTA [1993]) ainsi que l'IA (VERFAILLIE [1997], DUBOIS & PRADE [2002]) se sont beaucoup intéressées à la classification des problèmes en problématiques. Ainsi différentes classifications existent, et les frontières ne sont pas toujours très distinctes.

Pour que le processus de résolution soit couronné de succès, il est nécessaire que les problèmes d'aide à la décision soient pertinents pour les utilisateurs, et réduisent au maximum l'effort cognitif demandé. Cette pertinence va aussi dans le sens d'un élargissement des perspectives des décideurs : Les SIAD vont augmenter la créativité des utilisateurs en leurs donnant le support cognitif qui faisait défaut à leur intuition et à leurs connaissances (LÉVINE & POMEROL [1989, § 11.2]). Les problématiques techniques constituées sont souvent liées à différents processus de raisonnements (COURBON *et al*. [1994]) que l'homme peut effectuer pour progresser dans la recherche de solutions acceptables : le choix, la déduction, le raisonnement plausible, le raisonnement abductif, la révision des croyances, l'analogie, etc.

Dans la suite de cette section, nous décrivons les caractéristiques des problématiques d'aide à la décision, puis nous nous attardons sur quelques problématiques. Certaines sont utilisées dans le reste de ce document.

3.2.3.1. Axiomes et Problématiques d'Aide à la Décision : les Règles de Décision

Chaque problématique d'aide à la décision repose sur un certain nombre d'axiomes qui caractérisent les problèmes qui s'y rapportent. Certains de ces axiomes décrivent les propriétés liant les données d'un problème technique à ces résultats, formant des règles de décision. Une

[53] Voir aussi le site web du groupe de travail européen EUROGDSS : http://infolab.kub.nl/eurogdss/.

règle de décision est en quelque sorte une application qui spécifie, pour un problème d'aide à la décision donné, si un objet donné peut être considéré comme une solution réalisable et/ou satisfaisante de ce problème en une instance donnée. Une règle de décision peut être vu comme une caractérisation du problème d'appartenance (§ 2.1.1.1, p. 30) associé au problème d'aide à la décision.

Dans le reste de cette section, nous présentons succinctement quelques problématiques d'aide à la décision qui sont utilisées dans la pratique : la problématique d'aide au choix, les problématiques de tri et de rangement, la problématique technique de consistance, la problématique d'aide face à l'incohérence et la problématique technique dynamique. Intéressons-nous à la première. Qu'est-ce que la problématique d'aide au choix ?

3.2.3.2. Problématique d'Aide au Choix

La problématique d'aide au choix est l'une des plus étudiées, et admet comme cas particulier la problématique de l'optimisation simple (ROY [1985]). Cette problématique considère un ensemble S donné en extension ou en compréhension, appelé ensemble des solutions réalisables alternatives, et un ensemble de préférences *pref*(S) formulées sur S par différents acteurs.

La **problématique d'aide au choix** consiste à utiliser une procédure de sélection orientée vers la mise en évidence d'un sous-ensemble S' de S de solutions satisfaisantes (compte tenu des systèmes de valeurs *pref*(S) des acteurs). Ce sous-ensemble sera aussi restreint que possible en vue de justifier le « non-choix » du plus grand nombre possible de solutions[54].

L'identification du sous-ensemble S' à partir des préférences *pref*(S) n'est pas toujours directe et opérationnelle. Les problèmes d'aide à la décision étant entièrement automatisables, il est alors nécessaire d'ajouter de l'information pour globaliser et synthétiser l'information préférentielle. Les prémisses ajoutées sont appelées **axiomes de choix**. Par exemple, les problèmes d'optimisation simples reposent sur l'axiome de choix informant qu'une solution est satisfaisante si elle est au moins d'aussi bonne qualité que toutes les autres.

Cette caractérisation sert de base aux spécifications des problèmes automatisables apportant cette aide. Ainsi un problème Π aide au choix de solutions satisfaisantes pour *pref*(S) sur S, si son instance contient le couple (S, *pref*(S)) défini en extension ou non, et que son objectif est d'identifier un ensemble aussi réduit que possible (éventuellement réduit à une seule solution) de solutions jugées de meilleure qualité que les autres par *pref*(S), sans s'imposer d'aboutir à la mise en évidence d'une solution réputée optimale. Les problèmes d'aide au choix sont de la forme suivante :

> Étant donnés un couple (S, *pref*(S)) et un ensemble de propriétés remarquables (axiomes de choix), retourner un sous-ensemble de solutions dans S ayant ces propriétés, si un tel ensemble non vide existe, sinon retourner "non".

L'investigation réelle de ce sous-ensemble de solutions, se fera selon la nature de S (est-il donné en extension ou non ?). Ainsi, si S est donné en extension, une énumération des éléments de ce sous-ensemble peut être envisagée. Dans le cas contraire, une interaction avec l'utilisateur est généralement nécessaire[55]. Par la suite, la problématique d'aide au choix est étudiée dans deux cas particuliers :

(a) Le modèle préférentiel du problème est une relation binaire de préférence (§ 3.3.1) unique sur l'ensemble S des solutions (§ 3.4), et

[54] Extrait de BANA E COSTA [1993]. Pour une définition plus complète, voir ROY [1985, pp. 75-77] ou ROY & BOUYSSOU [1993].

[55] Cette communication de la solution n'est pas automatique. Le sous-ensemble identifié peut être retourné en compréhension, par exemple, un système d'inégalités linéaires si ce sous-ensemble est un polyèdre, etc.

(b) Le modèle préférentiel du problème correspond aux préférences de plusieurs individus pour lesquels les solutions de meilleur compromis doivent être caractérisées (§ 4.2 et chap. 6).

La définition de la problématique d'aide au choix identifie deux types d'axiomes de choix principaux. Les axiomes garantissant la qualité pour les protagonistes des solutions choisies, et les axiomes exprimant que l'information retournée doit être minimale en nombre d'éléments. Notre présentation des axiomes se calquera sur cette classification.

3.2.3.3. Problématiques de Tri et de Rangement

Ces problématiques sont spécifiques aux problèmes dont les solutions sont données en extension. Elles ont fait l'objet de beaucoup d'attention en AMCD.

La problématique de tri consiste à affecter les solutions potentielles à des catégories prédéfinies, en se fondant uniquement sur les caractéristiques propres aux solutions. Quelques exemples d'applications : le diagnostic médical, l'octroi de crédit, l'évaluation d'étudiants.

La problématique du rangement vise à établir un ordre de préférence sur des solutions, à partir de plusieurs critères. Elle est particulièrement utile pour sélectionner des étudiants à un concours, choisir des projets d'investissement dans la limite d'un budget donné.

Nous ne nous étendons pas plus sur ces problématiques dans ce document et renvoyons à ROY [1985], ROY & BOUYSSOU [1993].

3.2.3.4. Problématique Technique de la Consistance

Cette problématique a véritablement pris son essor avec les systèmes de contraintes dans les années 70. Elle considère un système de contraintes à partir duquel elle identifie un système de contraintes équivalent au premier, comportant moins de valeurs et/ou de combinaisons de valeurs des domaines ne conduisant à aucune solution réalisable pour le CSP.

Un problème de consistance consiste à modifier un système de contraintes de manière à ce qu'il vérifie un ensemble de propriétés appelées **axiomes de consistance**. Il existe plusieurs types de consistance :

- La **consistance directe** qui consiste à supprimer des combinaisons de valeurs contenues dans aucune solution réalisable (FREUDER [1982], [1985], COOPER [1989]) ;
- La **consistance inverse** qui consiste à supprimer des valeurs contenues dans aucune solution réalisable (FREUDER [1985], VERFAILLIE *et al*. [1999]) ;
- La **consistance relationnelle** (DECHTER & VAN BEEK [1995]).

Nous présentons ici succinctement la consistance inverse, les autres étant restées principalement dans la sphère théorique. Ce type de problèmes consiste à simplifier un système de contraintes, en supprimant uniquement des valeurs de domaines qui ne peuvent conduire à des solutions réalisables pour le problème de satisfaction de contraintes (CSP). De telles valeurs ne conduisant à aucune solution réalisable sont dites **inconsistantes**. Tout problème de consistance inverse vérifie les propriétés (axiomes de consistance) suivantes : l'ensemble des solutions réalisables ainsi que l'ensemble des contraintes sont inchangées (invariantes). De plus, chaque domaine de variables du système résultat est inclus dans le domaine du système initial. En d'autres termes, si on note $\mathcal{R} = (V, D, C)$ le système de contraintes initial et $\mathcal{R}' = (V, D', C')$ le système de contraintes résultat, alors $S_{CSP}(\mathcal{R}) = S_{CSP}(\mathcal{R}')$, $C' = C$ et $\emptyset \neq D'(v) \subseteq D(v)$ pour tout $v \in V$.

Voici deux problèmes très importants de cette classe :

Globale inverse consistance (GIC) : Étant donné un système de contraintes $\mathcal{R} = (V, D, C)$, trouver le système de contraintes $\mathcal{R}' = (V, D', C)$ tel que : $S_{CSP}(\mathcal{R}) = S_{CSP}(\mathcal{R}')$; $\emptyset \neq D'(v) \subseteq D(v)$ pour tout $v \in V$; et pour toute variable $v \in V$ et pour tout $d_v \in D'(v)$, il existe $x \in S_{CSP}(\mathcal{R})$ tel que $x(v) = d_v$. Sinon retourner "non".

Le problème de consistance GIC, ne conserve dans les domaines que les valeurs conduisant à une solution réalisable. Il possède un certain nombre de propriétés très intéressantes. Tout d'abord, $\mathcal{R}'$ est unique. De plus, si les domaines sont en extension, et qu'un tel algorithme polynomial est disponible pour un système de contraintes et tout système obtenu en restreignant les domaines de ce système, alors le problème CSP est polynomial. Ce problème de consistance se veut être la projection exacte de l'ensemble des solutions réalisables selon chaque variable du système de contraintes.

Borne-consistance : Étant donné un système de contraintes $\mathcal{R} = (V, D, C)$, dont les domaines sont totalement ordonnés et possédent un plus grand élément $\max(D(v))$ et un plus petit élément $\min(D(v))$ pour tout v, trouver le système de contraintes $\mathcal{R}' = (V, D', C)$ tel que : $S_{\mathrm{CSP}}(\mathcal{R}) = S_{\mathrm{CSP}}(\mathcal{R}')$; $\varnothing \neq D'(v) \subseteq D(v)$ pour tout $v \in V$; et que pour toute variable $v \in V$, il existe $x, y \in S_{\mathrm{CSP}}(\mathcal{R})$ telles que $x(v) = \min(D'(v))$ et $y(v) = \max(D'(v))$; sinon retourner "non".

Le problème de borne-consistance est surtout utilisé quand les domaines sont totalement ordonnés en compréhension. Lorsque les domaines du système de contraintes initial sont des intervalles, il garantit que les bornes de chaque intervalle sont contenues dans une solution réalisable.

Ces problèmes sont généralement aussi difficiles que le CSP sur le système initial. Par conséquent, des problèmes de consistance vérifiant des axiomes de consistance plus faibles sont utilisés pour éliminer une partie des valeurs inconsistantes. D'autres problèmes de consistance inverse sont donnés en Annexe A2.

Le vocabulaire pour désigner un algorithme résolvant un problème de consistance est très riche : **algorithme de filtrage**, **de propagation de contraintes**, **d'élagage**, **d'établissement d'un niveau de consistance**, **d'établissement d'un niveau de cohérence**.

Ces problèmes sont à la fois des vérificateurs de cohérence et des filtres intelligents d'information au sens du § 3.2.2.4 : Une certaine forme d'information stérile en terme de solution réalisable est supprimée.

Les processus de décision basés uniquement sur l'établissement d'un niveau de consistance sur le système de contraintes sont appelés **processus de maintien de la cohérence** (voir par exemple MARTINEZ & VERFAILLIE [1998]). Ce type de processus est un support informatique particulièrement bien adapté à la gestion réactive, comme l'auto-planification ou la gestion de projet. Les méthodes PERT et CPM[56] sont justement des cas particuliers de ces processus : Le problème traité considère un ensemble de tâches, et une relation de précédence entre ces tâches (antériorité de certaines tâches par rapport à d'autres). A chaque tâche sont associées une durée et une date de début variable, les domaines de valeurs des dates de début étant donnés. Le but est alors d'éliminer les valeurs pour les dates de début de tâches qui ne mènent à aucune solution réalisable et qui minimisent la durée totale du projet. En fonction des avancées du projet et des modifications conséquentes des données, le système réagit et supprime les valeurs incohérentes, inconsistantes.

Les problèmes de consistance ont été très sporadiquement étudiés en RO en plus des problèmes de gestion de projet[57].

---

[56] PERT = *Program evaluation and review technique* ; CPM = *Critical path method* (voir par exemple SAKAROVITCH [1984*b*]).

[57] Par exemple, la notion d'*ensemble de type* Δ dans ROY [1969 § I.B.5.b] définit une structure formée d'une intersection de contraintes analytiques sur des variables à valeurs dans des domaines totalement ordonnés (voir ROY [1970, section X.C]. Mais elle a donné naissance à une large recherche que 10 ans plus tard, avec la programmation par contraintes en IA, les renommant par la même occasion sous le nom de problèmes de satisfaction de contraintes continues.

3.2.3.5. Problématique d'Aide Face à l'Incohérence

Contrairement aux problématiques précédentes, la problématique d'aide face à l'incohérence est caractérisée par un ensemble de solutions réalisables mal défini. Les problèmes d'aide face à l'incohérence s'adressent aux problèmes combinatoires, où la réponse "non" lors d'inconsistance n'est pas suffisante. Un justificatif est nécessaire.

VERFAILLIE & LOBJOIS [1999] ont identifié des exemples de problèmes d'aide face à l'incohérence associés au CSP :

Explication de l'incohérence : Soit un système de contraintes $\mathcal{R}$ = (V, D, C) inconsistant. Une explication de l'inconsistance de $\mathcal{R}$ est un sous-ensemble C' de C tel que le problème $\mathcal{R}'$ = (V, D, C') soit inconsistant. Retourner une explication de taille minimale.

Relaxation cohérente : Soit un système de contraintes $\mathcal{R}$ = (V, D, C) inconsistant. Une relaxation de $\mathcal{R}$ est un sous-ensemble C' de C tel que le problème $\mathcal{R}'$ = (V, D, C') soit consistant. Retourner une relaxation de taille maximale.

En termes de complexité des problèmes, lorsqu'une explication de l'inconsistance de cardinalité minimale ou une relaxation cohérente de cardinalité maximale sont recherchées, VERFAILLIE & LOBJOIS [1999] montrent que la résolution de tels problèmes peut s'avérer très difficile. La résolution simultanée du problème initial (CSP), de l'explication de l'incohérence et de la relaxation cohérente, aboutit à des résultats en pratique très attrayants.

Les applications sont nombreuses car la plupart des modèles usuels ont été conçus de manière à retourner la réponse "non" lorsqu'il n'existe aucune solution réalisable. Des critiques virulentes de la part des utilisateurs, notamment des systèmes d'aide à l'élaboration de planning ont été justement prononcés à cet égard. "non" n'est assurément pas une réponse pour de tels problèmes. Il est souvent très difficile de se contenter d'un "non" lorsque le système n'a pas de solution réalisable. Plutôt avoir une solution irréalisable comme solution à notre problème, qu'une telle réponse. Par conséquent, le problème doit être reformulé. On ne cherche assurément pas à savoir si notre système est réalisable ! ! ! Sauf dans le cas d'un démarrage d'activité, mais même à ce niveau, une réponse plus élaborée comme le nombre d'employés nécessaire pour couvrir la charge, serait plus pertinent ; ou encore la révision à la baisse des niveaux minimaux de qualité de service. En général, ces problèmes sont formulés pour une gestion périodique (hebdomadaire, mensuelle) du personnel (voir § 1.1.3). En définitive, dans tous ces cas, la réponse négative est rarement la solution adéquate, la plus informative de la situation.

Une autre utilisation en élaboration de plannings concerne la gestion des incohérences dans le cadre de l'auto-planification : aide à la négociation.

Cette problématique s'intègre aussi dans une *aide dynamique à la révélation de préférences locales*. Lorsque les utilisateurs ont cru se passer de détailler leurs préférences, et n'ont usé que de contraintes minimales de qualité.[58] Cette problématique met en œuvre le raisonnement qu'est la révision des croyances.

Remarquons que la relaxation de contraintes dures pose la question fondamentale de la légitimité d'une telle action : Comment peut-on accepter de relaxer des contraintes exprimant la faisabilité du système ? Quelle est la signification des contraintes dures dans un système de contraintes ? La faisabilité d'une instance n'a pas toujours le même sens profond que la faisa-

[58] Cela illustre bien une nouvelle manière de s'attaquer à un problème : Ne spécifier que l'idéal, et n'instruire (de manière interactive donc) que les préférences pertinentes au fur et à mesure de la progression de la résolution. Cette manière de résoudre est adaptée aux problèmes comportant un modèle de préférences très complexe, et faute de temps, il est impossible d'instruire convenablement celui-ci. Il peut être utilisé même lorsque le modèle de préférence est supposé donné entièrement, sachant que les préférences peuvent évoluer au cours du temps. Comme font remarquer COURBON *et al*. [1994] : « une décision est un construit, la notion de décision automatique est antinomique ».

bilité du système réel étudié. Certaines contraintes dures ne sont que des indicateurs de niveau minimal de qualité ou des niveaux d'aspiration, et leur insatisfaction est préférée à une solution négative. Dans ce cas, il faut spécifier l'importance accordée à la suppression de chaque contrainte et combinaison de contraintes, les unes par rapport aux autres.

Pour d'autres références, nous renvoyons à JUSSIEN [1997], JUSSIEN & BOIZUMAULT [1997].

❖ Problématique de l'Aide face à l'Incohérence en Situation de Choix

La problématique de l'aide face à l'incohérence n'est valable que dans le cas des CSP. Mais lorsque le problème admet des préférences émises par les acteurs, il est nécessaire de coupler l'aide au choix et l'aide face à l'incohérence. D'où la problématique mixte de l'**aide face à l'incohérence en situation de choix**. Une formulation générale est :

> « trouver une solution réalisable effectuant un meilleur compromis entre les préférences des acteurs si le système est cohérent, sinon fournir une solution – non réalisable donc – avec une explication de son incohérence qui soit la plus petite possible en terme d'importance vis-à-vis des contraintes dures à relaxer et des préférences. »

Nous nous attarderons au § 5.3.4 sur cette problématique mixte dans le cadre d'un système de contraintes sur lequel le modèle de préférence est une relation binaire de préférence (§ 3.3.1).

3.2.3.6. Problématique Technique Dynamique

La problématique dynamique se préoccupe de trouver une solution réalisable à partir d'une solution donnée non réalisable pour une instance connue d'un problème. Dans le cadre des CSP, VERFAILLIE & SCHIEX [1995] et VERFAILLIE [1997] ont défini le problème de réparation de solution comme suit :

Réparation de solution : Étant donnés un système de contraintes $\mathcal{R}$, une solution x de $\mathcal{R}$, ainsi qu'un ensemble de contraintes à supprimer de $\mathcal{R}$ et un autre à ajouter à $\mathcal{R}$, trouver une solution réalisable qui soit la moins éloignée de x.

L'éloignement se mesure généralement avec la distance de Hamming, mais il existe d'autres possibilités (RÉGIN *et al*. [2001]). L'ajout d'une contrainte nécessite généralement un remaniement du problème, si le statut de la solution x est modifié (elle passe de solution réalisable à solution non réalisable). Le but est alors de rechercher une solution réalisable qui soit la plus proche de la solution initiale.

Comme l'aide face à l'incohérence, la réparation de solutions nécessite généralement une concertation des acteurs du système à ne pas négliger dans les applications réelles. Une étape d'acquisition d'information préférentielle est nécessaire entre la donnée du problème et sa résolution pratique.

❖ Problématique Technique Dynamique Qualitative

Si, de plus nous connaissons l'avis d'un ou de plusieurs individus sur les différentes solutions, alors le problème devient une problématique mixte, et le problème de réparation de solution s'énonce comme suit :

Réparation de solution qualitative : Étant donnés un système de contraintes $\mathcal{R}$, les préférences $pref(S_{CSP}(\mathcal{R}))$ d'un ensemble d'individus sur $S_{CSP}(\mathcal{R})$, une solution x de $\mathcal{R}$, ainsi qu'un ensemble de contraintes à supprimer et un autre à ajouter de $\mathcal{R}$, trouver une solution de bonne qualité pour $pref(S_{CSP}(\mathcal{R}))$ qui soit la moins éloignée de x.

L'ajout et la suppression de contraintes modifient le statut de la solution x :

- de solution réalisable, x devient une solution non réalisable ; ou bien,

– de solution de meilleure qualité (optimale par exemple), x devient une solution de qualité améliorable.

L'instance à résoudre dépend de la cause ayant créée l'invalidité de la solution x. Prenons l'exemple des plannings. Les préférences sont susceptibles d'évoluer différemment selon que la cause est due au planificateur lors de sa recherche de solutions proches de x, ou que la cause est une absence imprévue après la fixation du planning x depuis plusieurs jours. En effet, le contexte a changé. Le fait que les acteurs soient au courant et aient eu le temps de s'adapter au nouveau planning, change les donnes du problème.

DAGO & VERFAILLIE [1996] ont étudié le problème de réparation de solution qualitative dans le cas des systèmes de contraintes valuées (VCSP). Dans ce cadre nous retrouvons aussi la programmation linéaire avec les méthodes basées sur le simplexe dual, celles-ci étant utilisées lorsque la solution initiale n'est plus réalisable à cause d'un ajout de contraintes linéaires (SAKAROVITCH [1984*a*]).

3.2.4. Conclusion

Il existe bien d'autres problématiques : Par exemple celle liée à la robustesse (contenant l'analyse post-optimale et l'analyse de sensibilité)[59], celle liée à l'extraction de connaissance (§ 3.2.2.5) ou encore les problématiques mixtes.

Toutes ces problématiques peuvent conduire à une aide automatisée dans un processus de décision. Pour une présentation d'autres problématiques en AMCD, nous indiquons les documents de BANA E COSTA [1993], de ROY [1985].

La problématique décisionnelle de l'élaboration de plannings de personnel est celle du choix d'une unique solution (§ 1.3). Nous avons pris le parti de baser nos travaux sur la conception de **systèmes interactifs d'aide à l'élaboration de plannings**. Par conséquent, un certain nombre de problèmes d'aide à la décision seront traités. Ainsi, nous étudions :

– la problématique d'aide au choix dans un cadre général (§ 3.4 et 4.2) et dans le cadre des CSP (§5.3.3 et 7.1).
– la problématique technique de la consistance (§ 7.3 et 7.4) ;
– la problématique mixte de l'aide face à l'incohérence en situation de choix (§ 5.3.4) ; et
– la problématique mixte technique de la consistance qualitative, couplant la consistance au choix (§ 5.3.5).

Nous avons vu que les préférences jouent un rôle primordial dans un grand nombre de problématiques d'aide à la décision. Les modèles préférentiels de l'optimisation classique sont basés exclusivement sur les fonctions objectives (vrai-critères). Mais d'autres disciplines ont fait grand usage des préférences, et se sont penchées sur leur modélisation. Elles ont défini un nombre considérable de modèles plus adéquats aux situations auxquelles elles font face.

3.3. Modèles de Base des Préférences

Il existe aujourd'hui un grand nombre de structures, utilisées en TCS et en AMCD, plus adaptées à la modélisation des préférences que la fonction objective. Nous présentons dans cette section les différentes extensions du modèle de vrai-critère qui ont été entreprises, afin de mieux décrire l'information préférentielle.

Les remises en question de propriétés sous-jacentes à la fonction objective ont concerné les points suivants :

– élargissement du champ des jugements à d'autres situations que la préférence stricte, l'aversion stricte et l'indifférence ;

[59] Dans des contextes très différents voir DURAND [2000] et ROY & BOUYSSOU [1993] en AMCD, SAKAROVITCH [1984*a*] en programmation linéaire, GREENBERG [1998] pour les MIP.

– affaiblissement de la structure algébrique transitive de la préférence stricte et de l'indifférence ;
– remise en question de la logique bivalente sous-jacente aux choix ; i.e. face à la question : « x est-il meilleur que (resp. indifférent à) y ? » il est possible de répondre autre chose que oui ou non ;
– relaxation de la source uniquement binaire des choix globaux ; i.e. pour savoir si oui ou non, x est le meilleur élément, il peut ne pas être suffisant de connaître les réponses sur les couples d'éléments (x, y) pour tout y ;
– acceptation d'une multiplicité des points de vue incommensurables (ou en tout cas, partiellement comparables), interne aux choix individuels et collectifs, avec la notion de profil.

Ainsi, au § 3.3.1 nous présentons la relation binaire de préférence, laquelle concerne les deux premiers points. Au § 3.3.2 quelques généralisations de la relation binaire de préférence sont présentées, portant sur les trois premiers points. En AMCD, plusieurs généralisations de la fonction objective ont été proposées et sont exposées au § 3.3.3. Le § 3.3.4 est consacré aux relations non-binaires de préférence qui généralisent la relation binaire de préférence en intégrant le quatrième point. Contrairement aux quatre premiers, le cinquième point tient compte de la dimension collective des préférences. Le chapitre 4 lui est entièrement consacré.

Les propriétés des modèles supportant les préférences d'un individu sont appelés axiomes de cohérence des choix de cet individu[60].

3.3.1. Les Relations Binaires de Préférences

Nous avons généralement tendance à croire que pour émettre un choix, l'identification des solutions alternatives à un nombre est nécessaire, la comparaison des nombres impliquant directement celle des solutions. D'où l'idée de la structure de vrai-critère. Mais, au lieu d'utiliser un modèle aussi complexe – utilisation d'un morphisme – pour véhiculer de l'information préférentielle, pourquoi ne pas simplement considérer une relation binaire sur S véhiculant la même information ? C'est l'idée de la relation binaire de préférence.

3.3.1.1. La Relation Binaire

Avant de présenter ce modèle support d'informations préférentielles, voici quelques propriétés des relations binaires.

❖ Quelques propriétés

Rappelons qu'une **relation binaire** $\succcurlyeq$ sur un ensemble S quelconque est un sous-ensemble du produit cartésien $S \times S$. Nous noterons $x \succcurlyeq y$, lorsque le couple ordonné (x, y) est dans $\succcurlyeq$, et $\mathsf{non}[x \succcurlyeq y]$ dans le cas contraire :

$$x \succcurlyeq y \Leftrightarrow (x, y) \in \succcurlyeq \qquad \mathsf{non}[x \succcurlyeq y] \Leftrightarrow (x, y) \notin \succcurlyeq .$$

Une relation binaire $\succcurlyeq$ sur S est :
– **réflexive** si : $x \succcurlyeq x$, pour tout $x \in S$.
– **irréflexive** si : $\mathsf{non}[x \succcurlyeq x]$, pour tout $x \in S$.

[60] La cohérence des choix d'un individu est plus connue sous le nom de comportement rationnel ou rationalité de l'individu. Mais ce terme porte à confusion, comme le faisait remarquer SEN [1987]. Car jusqu'à la seconde partie du 20e siècle, le seul comportement rationnel admis, véhiculé par les fonctions objectives, suppose la transitivité des attitudes fondamentales. Et toute structure de préférence qui ne satisfait pas cette propriété étaient traités d'irrationnels et donc d'inacceptables. Le terme de rationalité limitée a par la suite, fait son apparition avec notamment les travaux de H. A. SIMON (§ 3.1.3). Mais ce terme est lui-même à connotation négative car un individu pleinement rationnel peut aboutir à de telles incohérences dans les propriétés du modèle représentant ses préférences.

- **symétrique** si : $x \succcurlyeq y \Rightarrow y \succcurlyeq x$, pour tout $(x, y) \in S \times S$.
- **antisymétrique** si : $x \succcurlyeq y \Rightarrow \text{non}[y \succcurlyeq x]$, pour tout $(x, y) \in S \times S$, avec $x \neq y$.
- **complète** (ou **totale**) si : $x \succcurlyeq y$ ou $y \succcurlyeq x$, pour tout $(x, y) \in S \times S$, avec $x \neq y$.
- **transitive** si : $x \succcurlyeq y$ et $y \succcurlyeq z \Rightarrow x \succcurlyeq z$, pour tout $x, y, z \in S$.
- une **relation d'équivalence** si elle est réflexive, symétrique et transitive.
- un **ordre partiel** si elle est réflexive, antisymétrique et transitive (OP).
- un **ordre complet** (ou **total**) si elle est réflexive, antisymétrique transitive et complète (OT).
- un **préordre partiel** si elle est réflexive et transitive (POP).
- un **préordre complet** (ou **total**) si elle est réflexive, complète et transitive (POT).

D'autres propriétés importantes des relations binaires seront données par la suite.

❖ Opérations sur les Relations Binaires

Étant données deux relations binaires $\succcurlyeq_1$ et $\succcurlyeq_2$ sur S, nous dirons que :

- $\succcurlyeq_1$ est incluse dans $\succcurlyeq_2$ si : $x \succcurlyeq_1 y \Rightarrow x \succcurlyeq_2 y$, pour tout $(x, y) \in S \times S$.
- $\succcurlyeq_2$ est une **extension** de $\succcurlyeq_1$ si pour tout $x, y \in S$,
  (1) $x \succcurlyeq_1 y \Rightarrow x \succcurlyeq_2 y$ ($\succcurlyeq_1$ est incluse dans $\succcurlyeq_2$)
  (2) $x \succcurlyeq_1 y$ et $\text{non}[y \succcurlyeq_1 x] \Rightarrow x \succcurlyeq_2 y$ et $\text{non}[y \succcurlyeq_2 x]$.
- $\succcurlyeq_1$ est le **complémentaire** de $\succcurlyeq_2$, et on note $\succcurlyeq_1 = \text{comp}(\succcurlyeq_1)$, si :
  $x \succcurlyeq_1 y \Leftrightarrow \text{non}[x \succcurlyeq_2 y]$, pour tout $(x, y) \in S \times S$.
- $\succcurlyeq_1$ est l'**inverse** de $\succcurlyeq_2$, si : $x \succcurlyeq_1 y \Leftrightarrow y \succcurlyeq_2 x$, pour tout $(x, y) \in S \times S$.
- $\succcurlyeq$ est l'**union** de $\succcurlyeq_1$ et $\succcurlyeq_2$, si : $x \succcurlyeq y \Leftrightarrow x \succcurlyeq_1 y$ ou $x \succcurlyeq_2 y$ pour tout $(x, y) \in S \times S$.
- $\succcurlyeq$ est l'**intersection** de $\succcurlyeq_1$ et $\succcurlyeq_2$, si :
  $x \succcurlyeq y \Leftrightarrow x \succcurlyeq_1 y$ et $x \succcurlyeq_2 y$ pour tout $(x, y) \in S \times S$.
- $\succcurlyeq$ est la **composition** de $\succcurlyeq_1$ par $\succcurlyeq_2$, si :
  $x \succcurlyeq z \Leftrightarrow x \succcurlyeq_1 y$ et $y \succcurlyeq_2 z$ pour tout $(x, y, z) \in S \times S \times S$.

Rappelons que toute relation binaire $\succcurlyeq$ sur un ensemble S fini peut être représentée par un graphe orienté simple $D = (S, A)$, avec : $x \succcurlyeq y \Leftrightarrow (x, y) \in A$ pour tout $(x, y) \in S \times S$. Par conséquent le vocabulaire utilisé pour les graphes orientés simples est valable pour les relations binaires sur un ensemble fini, et vice versa.

3.3.1.2. La Relation Binaire de Préférence : Définition et Propriétés

Une **relation binaire de préférence** *rbp*(S) sur un ensemble S est une relation binaire $\succcurlyeq$ réflexive sur S,

- véhiculant l'information préférentielle générique suivante :
  pour x, y deux éléments de S, $x \succcurlyeq y \Leftrightarrow x$ « est au moins aussi bon que » y ;
- identifiant quatre relations binaires par les définitions : Pour tout $(x, y) \in S \times S$,
  (a) $x \simeq y \Leftrightarrow x \succcurlyeq y$ et $y \succcurlyeq x$ (partie symétrique de $\succcurlyeq$)
  (b) $x \succ y \Leftrightarrow x \succcurlyeq y$ et non $[y \succcurlyeq x]$ (partie asymétrique de $\succcurlyeq$)
  (c) $x \parallel y \Leftrightarrow$ non $[x \succcurlyeq y]$ et non $[y \succcurlyeq x]$
  (d) $x \prec y \Leftrightarrow y \succcurlyeq x$ et non $[x \succcurlyeq y]$

  les quatre relations binaires possédant les propriétés structurelles et informationnelles minimales suivantes :
  - L'**indifférence** (noté : x est indifférent à y, ou encore $x \simeq y$) :
    propriétés : relation symétrique réflexive
    information préférentielle : il existe suffisamment d'arguments clairs et positifs justifiant une équivalence entre les deux solutions comparées.
  - La **préférence stricte** (noté : x est mieux que y, ou encore $x \succ y$) :
    propriétés : relation antisymétrique irréflexive

information préférentielle : il existe des raisons claires et positives pour préférer la solution x à la solution y.

– L'**aversion stricte** (noté : x est pire que y, ou encore $x \prec y$) :
  propriétés : relation antisymétrique irréflexive
  information préférentielle : il existe des raisons claires et positives pour rejeter la solution x par rapport à la solution y.

– L'**incomparabilité** (noté : x est incomparable à y, ou encore $x \parallel y$) :
  propriétés : relation symétrique irréflexive
  information préférentielle : il n'y a pas de raisons suffisantes qui justifient l'une des trois situations précédentes.

Dire qu'une relation binaire $\succcurlyeq$ sur un ensemble S, est une relation binaire de préférence $rbp(S)$ sur S, c'est accepter l'information préférentielle quelle véhicule, ainsi que celles des relations binaires induites via les relations (a) à (d). Dans ce cas, la relation binaire $\succcurlyeq$ véhicule toute l'information nécessaire à la formulation des préférences d'une relation binaire de préférence. Dans le reste de ce document, nous nous plions à cet usage, et nous employons indifféremment $\succcurlyeq$ et $rbp(\cdot)$ pour désigner une relation binaire de préférence.

❖ Quelques Propriétés Induites

Les relations binaires sous-jacentes à une relation binaire de préférence $\succcurlyeq$ respectent les propriétés suivantes :

- pour tout couple $x, y \in S$, il existe un unique $\alpha \in \{\succ, \simeq, \prec, \parallel\}$, tel que $x \,\alpha\, y$. (les relations binaires sous-jacentes forment une partition de $S \times S$) ;
- pour tout couple $x, y \in S$, $\quad x \succcurlyeq y \Leftrightarrow x \simeq y$ ou $x \succ y$.
- pour tout couple $x, y \in S$, $\quad x \parallel y \Leftrightarrow$ non $[x \simeq y]$ et non $[x \succ y]$ et non $[y \succ x]$.

Les preuves sont assez directes.

❖ Graphe Orienté et Relation Binaire de Préférence sur un Ensemble Fini

Lorsque S est fini, la relation binaire de préférence $\succcurlyeq$, comme toute relation binaire (§ 2.1.2), est représentable graphiquement par un graphe simple orienté $D = (S, A)$ sur S. Pour passer du graphe à la relation binaire de préférence et inversement, on pose :

$$(x, y) \in A \Leftrightarrow x \succcurlyeq y \,. \tag{3-3}$$

Cette relation définit une bijection entre l'ensemble des relations binaires sur S et l'ensemble des graphes orientés simples et réflexifs sur S. Les propriétés (a) à (d) – définissant les relations binaires $\succ$, $\simeq$, $\prec$ et $\parallel$ à partir de $\succcurlyeq$ – et l'équivalence (3-3) permettent d'identifier ces relations binaires sous-jacentes à des graphes partiels. La Figure 11 reprend schématiquement ces définitions.

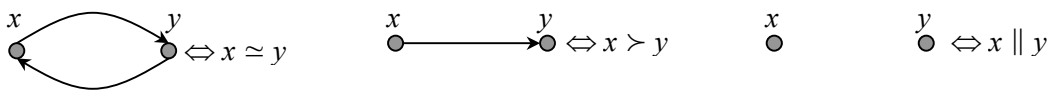


Figure 11 : Équivalence entre graphe orienté simple réflexif et relation binaire de préférence sur un ensemble fini.

❖ Du Vrai-Critère à la Relation Binaire de Préférence

Une fonction objective $g = (f, \geq)$ sur un ensemble S crée une relation binaire de préférence sur S, structurée en préordre total. En d'autres termes, l'incomparabilité n'est pas permise, et les trois autres relations restantes nécessitent d'être transitives. Seule la réflexivité est exigée

par la relation binaire de préférence. La littérature fourmille d'exemples où ces propriétés (transitivité et complétude) sont transgressées. Par exemple :

- Dans le contexte des votes (MARCHANT *et al*. [2000]), lorsqu'un votant compare deux candidats x, y sur lesquels il ne sait presque rien, à part leur nom et le fait qu'ils soient candidats, peut-il les déclarer pour autant indifférents ? Peut-il formuler une préférence pour un des candidats alors qu'il estime qu'il a trop peu d'information. Dans un tel contexte, l'incomparabilité serait nécessaire.
- Le cas très célèbre des tasses de thé[61], où la transitivité de l'indifférence est mise à mal : Supposons que l'on présente différentes tasses de thé à un individu dans l'ordre suivant : la première tasse sans sucre, la seconde avec un grain de sucre, la troisième avec 2 grains et ainsi de suite. Supposons en plus qu'il préfère le café sucré. La différence entre deux tasses successives est si faible qu'en le questionnant sur sa préférence, il répond l'indifférence. Par contre, si on lui présente la première tasse de thé (sans sucre) et la dernière (suffisamment bien sucré), il saura identifier la plus sucré comme son préféré. Ce qui contredit la transitivité de l'indifférence qui veut que la première et la dernière tasse soient indifférentes.

D'autres exemples peuvent être trouvés dans la littérature. Notamment, certaines préférences collectives obtenues avec des procédures de vote (comme la célèbre règle de la majorité) à partir de préférences individuelles transitives, induisent des intransitivités assez fréquemment, comme nous le verrons au § 4.2.1.4.

L'acceptation de propriétés aussi restrictives que celle du vrai-critère, limite par la même occasion l'information préférentielle qu'il est possible d'exprimer. Or, pour nos problèmes par exemple, une telle hypothèse n'a pas lieu d'être. Ces écarts aux propriétés structurelles du vrai-critère invalident par la même occasion l'évaluation numérique utilisée pour modéliser n'importe quelle information préférentielle. Nous verrons cependant au § 3.3.3 des modèles proposés en AMCD, définissant des structures plus faibles sur l'ensemble des solutions, et utilisant pourtant une évaluation cardinale.

❖ Quelques Propriétés Remarquables Supplémentaires pour les Relations Binaires de Préférence

Dans la littérature, différentes propriétés structurelles plus faibles que le préordre total, mais ne se limitant pas aux propriétés minimales, ont été envisagées. Voici quelques propriétés des relations binaires de préférence qui nous intéressera par la suite. Une relation binaire de préférence $\succcurlyeq$ est :

- **quasi-transitive** si : pour tout $x, y, z \in S$, $\quad x \succ y$ et $y \succ z \Rightarrow x \succ z$.
- **P-acyclique** si : pour $2 < t < +\infty$ et $x_1 \succ x_2 \succ \ldots \succ x_t \Rightarrow x_1 \succcurlyeq x_t$.
- **strictement P-acyclique** (sans circuit antisymétrique) si : pour $t > 2$ et $x_1 \succ x_2 \succ \ldots \succ x_t \Rightarrow$ non $[x_t \succ x_1]$.
- **un ordre d'intervalle** si pour tout $x, y, z, t \in S$, $\quad x \succ y \simeq z \succ t \Rightarrow x \succ t$.

Les relations binaires structures sont limitées aux quatre attitudes fondamentales de préférence stricte, aversion stricte, indifférence et incomparabilité. D'autres modèles sont apparus, généralisant ces structures, de manière à modéliser l'imprécision ou encore différents niveaux d'appréciation d'une attitude fondamentale. Ils sont présentés ci-dessous.

3.3.2. Préférences Binaires et Structures de Préférence

Certains jugements plus subtils, véhiculant de l'imprécision ou de la nuance, sont impossi-

[61] Présenté par Luce dans un article datant de 1956, cet exemple peut être trouvé dans VINCKE [1989], ou MARCHANT *et al*. [2000].

bles à formuler avec la relation binaire de préférence ou le vrai-critère. Par exemple, on peut vouloir[62] :

- apprécier de différentes manières, la préférence stricte,
- donner différents sens à l'incomparabilité : Sont-ils incomparables par manque d'information, informations confidentielles, informations conflictuelles ?
- graduer le passage de l'indifférence à la préférence stricte,
- représenter l'information préférentielle collective non opérationnelle,
- modéliser un concept comportant des zones d'équivoque, d'imprécision,
- etc.

En conséquence, la relation binaire de préférence a été généralisé de différentes manières. Nous présentons au § 3.3.2.2, un modèle générique faisant usage d'informations floues cardinales et ordinales. Elles font usage des notions d'attitude fondamentale et de structure de préférence (§ 3.3.2.1). Le § 3.3.2.3 est consacré aux préférences formulées à l'aide de la structure de préférence PC, qui est une autre façon de tenir compte de l'incertitude. Toutes ces préférences restent basées sur la comparaison de couples de solutions, et sont appelés par conséquent **préférences binaires**.

3.3.2.1. Attitudes et Structures de Préférence

❖ Définitions et Exemples

Les différents états, dans lesquels peut se trouver l'information préférentielle lorsque deux solutions sont comparées, sont appelés **attitudes fondamentales**. Chaque attitude fondamentale est une relation binaire sur un ensemble donné (si l'on utilise le vocabulaire de la théorie des ensembles) ou un prédicat binaire (si l'on utilise le vocabulaire de la logique et de la théorie des langages). L'ensemble des attitudes fondamentales est noté *AF*.

Par exemple, les quatre relations binaires d'indifférence, préférence stricte, aversion stricte et incomparabilité forment l'ensemble *AF*(*RBP*) des attitudes fondamentales de la relation binaire de préférence : $AF(RBP) = \{\succ, \simeq, \prec, \|\}$. La relation binaire réflexive $\succcurlyeq$ est une attitude de la relation binaire de préférence. Nous utiliserons indifféremment la notation $\{\succ, \simeq, \prec, \|\}$ ou {mieux, indifférent, pire, incomparable} pour désigner les attitudes fondamentales de la relation binaire de préférence.

Autre exemple avec la **relation de surclassement** (*outranking relation*, ROY & BOUYSSOU [1993]). Très étudiée en AMCD, cette relation base ses jugements sur un ensemble de solutions uniquement sur les attitudes fondamentales $\{\succcurlyeq, \preccurlyeq, \|\}$.

D'autres exemples sont donnés au paragraphe suivant.

Notons $\mathcal{PR}(AF)$ l'ensemble des sous-ensembles de *AF*, différents de l'ensemble vide et de *AF* : $\mathcal{PR}(AF) = \mathcal{P}(AF) \setminus \{\emptyset, AF\}$. Les éléments de $\mathcal{PR}(AF)$ sont appelés **attitudes**[63].

Dans le cadre de la relation binaire de préférence, nous utiliserons indifféremment la notation "surclasse" et $\succcurlyeq$ (resp. "sousclasse" et $\preccurlyeq$) pour désigner l'attitude $\{\succ, \simeq\}$ (resp. $\{\prec, \simeq\}$). Cette notation vient directement de l'équivalence qui existe entre la relation binaire de préférence et la relation de surclassement.

Les attitudes $\mathcal{PR}(AF)$ munies de propriétés structurelles, appelées **axiomes de cohérence**, forment une **structure de préférence** *SP*. Les axiomes de cohérence d'un ensemble d'attitudes peuvent être classés selon différents critères.

---

[62] Pour une analyse plus approfondie sur la nécessité d'utiliser des modèles étendus, voir PERNY [1992], TSOUKIÀS [1999].

[63] Une « attitude » est aussi appelée dans la littérature « code » (BELMANDT [1993], MOULEN *et al.*[1999]), ou « situation » (ROY & BOUYSSOU [1993]).

Les axiomes plus fréquents sont la symétrie et la transitivité. Considérons l'ensemble des attitudes $\mathcal{PR}(AF)$, une structure de préférence possède une **cohérence de** :

- **réflexivité**, lorsqu'il existe une attitude $\alpha \in \mathcal{PR}(AF)$ vérifiant :

  $$x \, \alpha \, x \qquad \text{pour tout } x \in S.$$

  Cette attitude est notée *ref*.
- **symétrie**, lorsque pour toute attitude $\alpha_1 \in \mathcal{PR}(AF)$, il existe une attitude $\alpha_2 \in \mathcal{PR}(AF)$ vérifiant :

  $$x \, \alpha_1 \, y \Rightarrow y \, \alpha_2 \, x \qquad \text{pour tout } x, y \in S, \text{ et } x \neq y.$$

  Notons *INV* la bijection de $\mathcal{PR}(AF)$ dans $\mathcal{PR}(AF)$ associée[64].
- **transitivité**, si pour tout couple d'attitudes (α_1, α_2) il existe un sous-ensemble d'attitudes noté $AP(\alpha_1, \alpha_2)$, minimum par l'inclusion, vérifiant :

  $$[x \, \alpha_1 \, y \text{ et } y \, \alpha_2 \, z] \text{ alors } [\text{il existe } \alpha_3 \in AP(\alpha_1, \alpha_2) \text{ tel que : } x \, \alpha_3 \, z]$$
  $$\text{pour tout } x, y, z \in S, \text{ et } x \neq y \neq z.$$

  Notons *T* la multi-application de $(\mathcal{PR}(AF))^2$ dans $\mathcal{P}(AF)$ associée. Remarquons que nous pouvons avoir : $T(\alpha_1, \alpha_2) = \emptyset$, ce qui indique que tout couple d'éléments (x, y) de *S* vérifiant $x \neq y$ et $x \, \alpha_1 \, y$ ne peut être suivi d'un couple (y, z) vérifiant $y \, \alpha_2 \, z$ et ce quelque soit $z \in S \setminus \{x, y\}$.

Donnons quelques exemples :

- la relation binaire de préférence possède une cohérence de symétrie INV_{RBP} décrite par le Tableau 4. Par contre elle ne possède aucune restriction sur les attitudes possibles mettant en jeu trois solutions (cohérence de transitivité). L'attitude réflexive est l'indifférence : ref_{RBP} = indifférent. Nous obtenons alors la structure de préférence :

  $$SP(RBP) = (AF(RBP), ref_{RBP}, INV_{RBP}).$$

α	indifférent	mieux	pire	incomparable
$INV_{RBP}(\alpha)$	indifférent	pire	mieux	incomparable

Tableau 4 : Cohérence de symétrie admise par les relations binaires de préférence.

- Le préordre total, lui, possède :
  - une cohérence de symétrie INV_{POT} obtenue en restreignant INV_{RBP} à l'indifférence, la préférence stricte et l'aversion stricte.
  - une cohérence de transitivité T_{POT} décrit par le Tableau 5.

  La structure de préférence associée à un préordre total est :

  $$SP(POT) = (\{\text{indifférent, mieux, pire}\}, ref_{RBP}, INV_{POT}, T_{POT}).$$

T_{POT}	indifférent	mieux	pire
indifférent	{indifférent}	{mieux}	{pire}
mieux	{mieux}	{mieux}	$AF(g)$
pire	{pire}	$AF(g)$	{pire}

Tableau 5 : Propriété de cohérence de transitivité des préordres totaux.

D'autres axiomes de cohérence des attitudes peuvent s'exprimer en termes relationnels sur $\mathcal{PR}(AF)$, comme la P-acyclicité, etc. Pour une liste plus exhaustive d'exemples de représentation de relations binaires de préférence possédant des propriétés de cohérence particulières, voir BELMANDT [1993] et MOULEN *et al*. [1999].

En définitive, différentes préférences binaires peuvent être décrites via les structures de

[64] Remarquons le lien existant avec l'opérateur inverse des relations binaires.

préférence[65].

En conséquence, les préférences effectives *pref*(S) sur un ensemble S prennent la forme d'une application p de $S \times S$ dans AF, associant à chaque couple de solutions l'attitude adoptée. p est appelée **fonction de préférence** sur S. En définitive, ce modèle des préférences sur S, est un triplet[66] formé de l'ensemble des solutions, d'une structure de préférence et d'une fonction de préférence :

$$pref(S) = (S, SP, p).$$

L'existence de la fonction de préférence assure la partition de l'ensemble $S \times S$ par les attitudes fondamentales.
Nous notons indifféremment, pour toute attitude fondamentale $\alpha \in \mathcal{PR}(AF)$, pour tout $x, y \in S$, $p(x, y) = \alpha \Leftrightarrow x\ \alpha\ y$ pour désigner que l'attitude adoptée lors de la comparaison de x et y est α.

Ainsi, une relation binaire de préférence *rbp*(S) (ou encore $\succcurlyeq$) sur un ensemble de solutions potentielles S, peut être aussi représentée par le triplet (S, $SP(RBP)$, p) formé de l'ensemble des solutions S, de la structure de préférence $SP(RBP)$ ={{indifférent, mieux, pire, incomparable}, INV_{RBP}}, et d'une fonction de préférence p associant à chaque couple de solutions une attitude fondamentale telle que :
Pour tout $x, y \in S$, $p(x, y) = \alpha \Leftrightarrow x\ \alpha\ y$ α étant une attitude fondamentale de $AF(RBP)$.

Par la suite, nous noterons $\mathcal{R}$(RBP, S), $\mathcal{R}$(POT, S), $\mathcal{R}$(POP, S), $\mathcal{R}$(OT, S), $\mathcal{R}$(OP, S), l'ensemble respectivement des relations binaires de préférences, préordres totaux, préordres partiels, ordres totaux et ordres partiels sur l'ensemble S.

Dans les deux sections qui suivent, nous présentons deux modèles de préférence, dont les structures de préférence sous-jacentes sont obtenues à partir d'une autre structure de préférence, afin de tenir compte de l'incertitude. Le premier représente l'incertitude par une information quantitative, numérique, alors que le second utilise une information qualitative, en remettant en question la logique bivalente des relations binaires de préférence.

3.3.2.2. Relations Binaires de Préférences Valuées

❖ Définition et Interprétations

Les logiques usuelles d'appréciation d'une proposition sont binaires, c'est-à-dire, qu'elles conduisent à la réponse « oui » ou « non ». La relation binaire de préférence valuée permet de passer à un autre statut, en nuançant ces deux extrêmes par des degrés d'adéquation.

Étant donnés un ensemble de solutions S et une attitude α, une **relation (binaire) de préférence valuée** μ_α sur S s'appuyant sur le prédicat α est une application de $S \times S$ dans [0, 1], qui à tout couple de solutions (x, y) associe un réel $\mu_\alpha(x, y)$. Le réel $\mu_\alpha(x, y)$ indique le degré de validité (niveau de crédibilité) de la proposition $x\ \alpha\ y$.

[65] Pour des démonstrations informelles de cette affirmation voir MOULEN *et al.* [1999], ou partielles voir SEN [1970]. Notons qu'Amartya SEN ne parle pas de cohérence de transitivité, mais cette cohérence des préordres est équivalente aux propriétés PP, IP, II et PI qu'il utilise.
[66] Cette structure a été présentée par BELMANDT [1993] sous le nom d'**espace préférencié**. Voir aussi MOULEN *et al.* [1999].

Une relation de préférence valuée ajoute du « flou » dans l'appréciation d'une proposition préférentielle[67]. Ainsi, lorsque le prédicat α correspond à la préférence stricte, c'est-à-dire α = "mieux", alors $\mu_{\text{mieux}}(x, y)$ représente le degré avec lequel x est strictement préférée à y, pour tout $x, y \in S$.

Le degré de validité $\mu_\alpha(x, y)$ peut aussi être décrit par le fait que l'ordre total canonique (naturel) sur [0, 1] définit un préordre total sur $S \times S$; Une relation de préférence valuée peut donc être considérée comme une famille emboîtée de sous-ensembles de couples de solutions. Nous arrivons alors à la notion de λ-coupe.

Considérons une relation de préférence valuée μ_α sur S, et un réel $\lambda \in [0, 1]$. Une **λ-coupe** de μ_α est l'ensemble $SC^{\alpha}_{\geq\lambda} = \{(x, y) \in S \times S$ tel que : $\mu_\alpha(x, y) \geq \lambda\}$.

La relation de préférence valuée μ_α peut être directement utilisée pour exprimer des préférences sur S. La structure de préférence associée à μ_α est alors notée $SP(\mu_\alpha)$, l'ensemble des attitudes fondamentales étant [0, 1]. Les axiomes de cohérences de $SP(\mu_\alpha)$ se déduisent directement de μ_α. Ainsi, $SP(\mu_\alpha)$ possède une cohérence de symétrie, si et seulement si pour chaque attitude $\lambda_1 \in [0, 1]$ il existe une attitude $\lambda_2 \in [0, 1]$ telle que :

$$\mu_\alpha(x, y) = \lambda_1 \Rightarrow \mu_\alpha(y, x) = \lambda_2 \qquad \text{pour tout } x, y \in S, \text{ et } x \neq y.$$

La traduction se fait aussi de la même manière si la structure de préférence $SP(\mu_\alpha)$ possède une cohérence de transitivité. D'autres propriétés de cohérence sur μ_α sont présentés dans PERNY [1992, chap. 3].

❖ Relation de Préférence Valuée Multidimensionnelle

Généralement, une relation de préférence valuée sert à nuancer les attitudes AP non satisfaisantes d'un premier modèle donné. Par conséquent, une seule relation de préférence valuée n'est pas suffisante lorsque l'ensemble d'attitudes AP de référence n'est pas réduit à l'attitude α et à son complémentaire $\text{comp}(\alpha)$. D'autres attitudes peuvent admettre des degrés d'appréciation.

Par exemple, connaître le degré de validité de la proposition « $x \succ y$ » n'implique pas que l'on connaisse celui de « $x \simeq y$ ». La relation binaire de préférence a été conçue de manière à ce que l'assertion « $x \succ y$ » exclut les autres assertions. Mais cette propriété n'est plus valable, car de nouvelles attitudes sont créées pour nuancer les premières.

Considérons donc un ensemble d'attitudes fondamentales AF, et étudions le nouveau modèle de préférence sur S qui peut être construit. Supposons que l'on ait une relation de préférence valuée μ_α, par attitude $\alpha \in \mathcal{PR}(AF)$. La nouvelle structure de préférence associée à $\{\mu_\alpha$ tel que : $\alpha \in \mathcal{PR}(AF)\}$ est notée $SP_v(AF)$. Le nouvel ensemble d'attitudes est alors $AP_v(AF) = [0, 1]^{\text{card}(\mathcal{PR}(AF))}$, associant à chaque couple de solutions, une attitude comportant $\text{card}(\mathcal{PR}(AF))$ composantes.

L'ordre total canonique sur [0, 1] induit une famille d'ordres partiels sur $AP_v(AF)$, chaque ordre étant indexé par une attitude de $\mathcal{PR}(AF)$. Ainsi, en notant $\succcurlyeq_\alpha$ la relation binaire de préférence engendrée par ([0, 1], $\geq$) sur $AP_v(AF)$ par l'intermédiaire du prédicat α, nous obtenons : pour tout $\alpha \in \mathcal{PR}(AF)$, pour tout $\beta_1, \beta_2 \in AP_v(AF)$,

$$\beta_1 \succcurlyeq_\alpha \beta_2 \Leftrightarrow \beta_1(\alpha) \geq \beta_2(\alpha) \quad \text{et} \quad \beta_1(\alpha_1) = \beta_2(\alpha_1) \quad \text{pour tout } \alpha_1 \in \mathcal{PR}(AF) \setminus \{\alpha\}.$$

$\beta_i(\alpha)$ étant la composante de l'attitude valuée β_i associée à l'attitude α.

La preuve que $\succcurlyeq_\alpha$ est un préordre partiel sur $AP_v(AF)$ est assez immédiate.

---

[67] Une **relation binaire floue** RBF sur S associée à μ_α correspond au graphe de l'application μ_α, c'est à dire :

$RBF = \{((x, y), \mu_\alpha(x, y))$ tel que : $(x, y) \in S \times S\}$. Pour un approfondissement nous renvoyons à PERNY [1992, chap. 3].

En terme de propriétés minimales de cohérence de $SP_v(AF)$ que pouvons-nous dire ? Dans le cas général l'ensemble $AP_v(AF)$ ne possède aucune structure minimale, hormis la famille d'ordres partiels. Maintenant, dans les modèles utilisés dans la littérature, beaucoup d'hypothèses sont émises, impliquant des redondances entre les relations de préférence valuées. La suppression des relations de préférence valuées redondantes permet alors de réduire considérablement la quantité d'informations nécessaires. Elle permet aussi d'obtenir des propriétés de cohérence pour la structure de préférence $SP_v(AF)$.
Ainsi, la relation d'inclusion sur $\mathcal{PR}(AF)$ induit souvent des liens entre les relations de préférences valuées associées.
Par exemple, le complémentaire d'une attitude α est généralement supposé décroître lorsque α croit et inversement. Par conséquent, il existe une fonction décroissante N_α définie sur [0, 1] à valeur dans [0, 1] vérifiant :

$$\mu_{\mathrm{comp}(\alpha)}(x, y) = N_\alpha(\mu_\alpha(x, y)), \qquad \text{pour tout } x, y \in S.$$

Dans ce cadre, N_α est appelé opérateur de négation.
Prenons un exemple : Considérons l'attitude α = « pire ». Le niveau d'appréciation de la proposition « x est au moins aussi bon que y » peut être inversement proportionnel au niveau d'appréciation de « x est pire que y ». Ce qui se traduit par : $\mu_{\succcurlyeq}(x, y) = 1 - \mu_{\prec}(x, y)$ pour tout $x, y \in S$, définissant alors $N_\alpha(\lambda) = 1 - \lambda$.
Il est bien entendu possible de valider d'autres opérateurs unaires, établissant un lien fonctionnel entre les degrés de validité de deux attitudes α et β. Par exemple, il peut exister un lien fonctionnel entre la préférence stricte et l'indifférence, supportant la question « x est préférable ou plutôt indifférent à y ? ».
D'autres relations peuvent être acceptées, comme les opérateurs binaires de normes triangulaires et co-normes triangulaires sur [0, 1].

Pour des développements plus conséquents, voir par exemple FODOR & ROUBENS [1994] ou PERNY [1992].

3.3.2.3. Structure de Préférence PC

Au lieu de nuancer l'affirmation et la négation d'une proposition par le concept de degré d'adéquation, établissant forcément un ordre total du moins crédible au plus crédible, il est possible d'utiliser plusieurs concepts intermédiaires incommensurables comme l'incertitude, ou l'inconsistance. Dans cet ordre d'idée, TSOUKIÀS & VINCKE [1995] ont présenté la **structure de préférence PC** (acronyme de *Partial Comparability*).

Cette structure de préférence, basée sur la structure de préférence des relations binaires de préférence $SP(RBP)$, fait usage du couple d'attitudes $\{\succcurlyeq, \preccurlyeq\}$ pour concevoir une structure de préférence.

A la proposition « $x \succcurlyeq y$ » est associée, non plus deux réponses possibles {oui, non}, mais quatre {oui, non, informations insuffisantes, informations contradictoires}. Les réponses informations insuffisantes et informations contradictoires permettent de tenir compte de l'hésitation. Cette hésitation peut être attribuée soit au manque d'information soit à l'inconsistance de l'information (TSOUKIÀS [1999]). On parle alors de **logique tétravalente** au lieu de logique bivalente.
L'hésitation est aussi prise en compte dans la réponse à la proposition « $x \preccurlyeq y$ ». En définitive, la comparaison d'une solution x à une solution y conduit à 16 attitudes fondamentales différentes induites par le produit cartésien des réponses aux questions « $x \succcurlyeq y$? » et « $x \preccurlyeq y$? » :

$$AF_{pc} = \{\text{oui, non, informations insuffisantes, informations contradictoires}\}^2.$$

Les propriétés de cohérence de la structure de préférence PC ont été choisies de manière à étendre la structure $SP(RBP)$. Ainsi, l'indifférence correspond à l'attitude (oui, oui),

l'incomparabilité est assimilée à l'attitude (non, non), la préférence stricte à l'attitude (oui, non) et l'aversion stricte à (non, oui).
Pour un approfondissement voir les références précitées.

3.3.2.4. Préférences entre Écarts de Préférences

La relation de préférence valuée peut être généralisée au cas non numérique, donnant naissance aux **préférences entre écarts de préférences** (ou **intensité de préférence**).

Ainsi, au lieu d'associer à chaque proposition un nombre, la comparaison est effectuée uniquement avec la relation binaire sous-jacente :

Étant donné un ensemble d'attitudes $AP \subseteq \mathcal{PR}(AF)$, une relation binaire de préférence $rbp(S \times S, \alpha)$, définie sur l'ensemble des couples de solutions, permet d'émettre des comparaisons entre les propositions $x\ \alpha\ y$, et ce quelque soit $\alpha \in AP$. Nous obtenons donc des propositions du type :

$$(x, y)\ \beta_\alpha\ (z, t), \qquad \text{avec } \alpha \in AP \text{ et } \beta \in \mathcal{PR}(AF(RBP))$$

exprimant par exemple :

« La préférence de x à y est-elle au moins aussi crédible que celle de z à t ? »

lorsque α est la préférence stricte, et β la relation $\succcurlyeq$.

Étant donné qu'il n'existe aucune échelle absolue support de la relation $\succcurlyeq$, comment se ramener à une comparaison sur S ? L'axiome suivant est souvent admis, lorsque $AP \subseteq \mathcal{PR}(AF(RBP))$:

$$(x, y)\ \beta_\alpha\ (z, z) \Leftrightarrow x\ \beta\ y.$$

Pour des exemples d'utilisation voir ROY & BOUYSSOU [1993].

Dans le prochain paragraphe, nous présentons les modèles basés sur une évaluation numérique mais imposants des propriétés structurelles à la relation de préférence sous-jacente, moins forte que celle du vrai-critère.

3.3.3. Fonction-Critères, Graduations

En AMCD, la notion de vrai-critère a été étendue de manière à repousser les limites de la modélisation de l'information préférentielle tout en conservant une évaluation numérique. Ainsi la notion de fonction-critère a été introduite.

3.3.3.1. Les Fonction-Critères

La structure classique utilisée en optimisation pour véhiculer une information préférentielle individuelle ou collective est la fonction objective (ou vrai-critère) présentée au § 2.1.1.1. Elle fait partie des fonction-critères.

Formellement, une **fonction-critère** g sur l'ensemble des solutions S est un couple (f, op) formé :

- d'une fonction f à valeurs réelles définie sur l'ensemble S, et
- d'une relation d'ordre total op usuelle de l'ensemble des réels : $\mathsf{op} \in \{\leq, \geq\}$;

autorisant la définition d'une comparaison $\succcurlyeq_g$ des solutions de S vérifiant la propriété suivante :

$$\text{pour tout } (x, y) \in S \times S,\ \ f(x)\ \mathsf{op}\ f(y) \Rightarrow x \succ_g y \ \text{ ou } \ x \simeq_g y\ ; \qquad (3\text{-}4)$$
$$\text{et} \quad f(x) = f(y) \Rightarrow x \simeq_g y.$$

Le réel $f(x)$ est appelé la **performance** (ou **score** ou **valeur**) de la solution x selon le critère g. La relation op permet de désigner l'**objectif** fixé, c'est à dire la **maximisation** ou la **minimisation** de la fonction f selon que op soit $\geq$ ou $\leq$. Les propriétés décrites par (3-4) de $\succcurlyeq_g$ ne garantissent pas son unicité (il n'est pas possible de trancher entre la préférence stricte et

l'indifférence de la première partie de (3-4). Elle n'informe que sur certaines propriétés minimales de $\succcurlyeq_g$, comme : la relation est complète et réflexive. Et lorsque S est fini, elle vérifie la proposition suivante :

Proposition 1 (fonction-critère et P-acyclicité) : Une fonction-critère g sur un ensemble fini S, définit une relation binaire de préférence sur S réflexive, complète et P-acyclique.

Preuve : Supposons que la relation n'est pas P-acyclique. Par conséquent, il existe une séquence finie $x_1, \ldots, x_t$ d'éléments disjoints de S, avec $2 < t < +\infty$ formant un circuit de préférences strictes. Les valeurs de ces éléments sont disjointes et totalement ordonnées. Cet ordre total sur les valeurs implique alors une incohérence avec le circuit stricte.

Remarquons qu'à toute fonction-critère à minimiser $g = (f, \leq)$, correspond une fonction-critère à maximiser $g' = (f', \geq)$, avec $f' = -f$. Par conséquent, la fonction-critère est souvent assimilée à la fonction de performance associée à l'objectif de maximisation. Nous ferons de même dans la suite de ce document.

D'autres axiomes sont nécessaires pour garantir l'unicité de $\succcurlyeq_g$. Voici quelques fonction-critères remarquables :

3.3.3.2. Le Vrai-Critère

Dans les modèles préférentiels traditionnels, d'autres conditions sont implicitement admises, définissant ainsi le **vrai-critère**[68]. Ces conditions imposent que l'inégalité stricte sur les rationnels implique la préférence stricte sur S. Il en est de même de l'égalité : l'égalité sur les rationnels implique l'indifférence sur S.

Formellement, un **vrai-critère** g sur l'ensemble des solutions S est une fonction-critère (f, op) vérifiant :

$$\text{pour tout } (x, y) \in S \times S, \quad [f(x) \ \mathsf{op}\ f(y) \text{ et } f(x) \neq f(y)] \Rightarrow x \succ_g y.$$

La relation binaire $\succcurlyeq_g$ est alors unique et possède une structure de préordre total (ROY [1985]). L'ensemble des vrai-critères sur S est noté $\mathcal{U}(S, \quad)$. Cette structure permet d'identifier les meilleures solutions de S selon g, à une valeur unique, appelée **valeur optimale**.

3.3.3.3. Le Critère d'Intervalle et le Quasi-Critère

Autre fonction-critère remarquable : Le critère d'intervalle nécessite une donnée supplémentaire : un **seuil d'indifférence** variable q, situant une frontière, en termes de distance dans l'espace des rationnels, entre l'inégalité stricte et l'indifférence.

Formellement, un **critère d'intervalle** g sur l'ensemble des solutions S est un couple (f, q) où :

- le couple $(f, \geq)$ est une fonction-critère, et
- q est une fonction réelle – définie sur l'image par f de l'ensemble S – à valeur réelle, et vérifiant :

  (a) pour tout $x \in S$, $q(f(x)) \geq 0$,

  (b) pour tout $(x, y) \in S \times S$, $f(x) > f(y) + q(f(y)) \Rightarrow x \succ_g y$, et

  (c) pour tout $(x, y) \in S \times S$, $\left.\begin{array}{l} f(x) \leq f(y) + q(f(y)), \text{ et} \\ f(y) \leq f(x) + q(f(x)) \end{array}\right\} \Rightarrow x \simeq_g y.$

La relation binaire $\succcurlyeq_g$ est alors unique et possède une structure d'ordre d'intervalle total (ROY [1985]). Dans le cas où le seuil d'indifférence est constant ($q(f(x))$ = constante, pour tout

[68] Le vrai-critère correspond aussi à la notion d'**indicateur** utilisé en sciences économiques, ou celle de **fonction objective**.

$x \in S$), la fonction-critère est appelée **quasi-critère**.

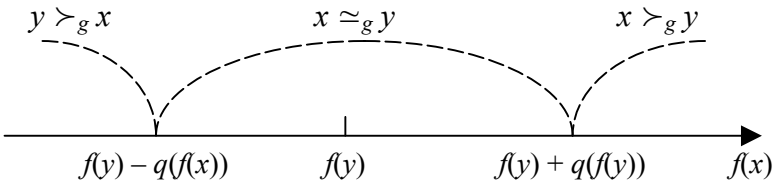


Figure 12 : Modèle du critère d'intervalle.

A partir d'une relation binaire sur S possédant les propriétés requises, quand pouvons-nous dire qu'il existe une fonction-critère ?

3.3.3.4. Le Pseudo-Critère

Jusqu'à maintenant, nous avons présenté des fonction-critères basées sur un ensemble d'attitudes fondamentales formé de l'indifférence, de la préférence stricte et de l'aversion stricte. En AMCD, un autre type de fonction-critère à vu le jour. Du nom de **pseudo-critère**, cette fonction-critère permet de nuancer le passage de la préférence ou aversion stricte à l'indifférence, avec les attitudes fondamentales intermédiaires de préférence faible et aversion faible. Leur prise en compte nécessite une donnée supplémentaire à celle du critère d'intervalle : le **seuil de préférence**. Nous ne présenterons pas cette structure ici. Le lecteur intéressé consultera ROY [1985], VINCKE [1989], ROY & BOUYSSOU [1993]. Bien entendu, un tel modèle peut être généralisé à des structures de préférence induisant plusieurs seuils de préférence.

3.3.3.5. Au Delà de la Fonction-Critère

Nous avons vu précédemment que la fonction-critère nécessite une structure de préférence admettant des propriétés structurelles minimales P-acycliques. Ces dernières années, des travaux ont été entrepris sur la modélisation d'information préférentielle ayant une cohérence des choix, plus faible que les fonction-critères, mais utilisant une formulation analytique. Ainsi, si Φ est une fonction de $S \times S$ dans quelconque, nous pouvons associer la relation binaire de préférence $\succcurlyeq_\Phi$ comme suit :

$$\text{pour tout } (x, y) \in S \times S, \quad \Phi(x, y) \geq 0 \Rightarrow x \succcurlyeq_\Phi y.$$

cette relation définissant entièrement les situations fondamentales d'indifférence, de préférence stricte et d'incomparabilité :

$x \simeq_\Phi y \Leftrightarrow x \succcurlyeq_\Phi y$ et $y \succcurlyeq_\Phi x$.
$x \succ_\Phi y \Leftrightarrow x \succcurlyeq_\Phi y$ et non $[y \succcurlyeq_\Phi x]$.
$x \parallel_\Phi y \Leftrightarrow$ non $[x \succcurlyeq_\Phi y]$ et non $[y \succcurlyeq_\Phi x]$.

Comme Φ est quelconque, la relation binaire de préférence $\succcurlyeq_\Phi$ ne possède aucune propriété particulière. La réciproque nécessite des conditions particulières. Pour un approfondissement, voir BOUYSSOU & PIRLOT [1997].

3.3.3.6. Axiomes de Mesurabilité, Graduations

Il est quelques fois très instructif de savoir si l'ordre induit par les valeurs et/ou les écarts entre valeurs doivent être pris en considération dans les décisions. Il en est fait usage en économie du bien-être (avec les axiomes de mesurabilité : SEN [1970, chap. 9 & 9*], D'ASPREMONT & GEVERS [2002]) et en AMCD (ROY [1985], ROY & BOUYSSOU [1993]) pour asseoir les décisions collectives entre plusieurs individus.

Dans un tel cadre, pour une fonction-critère $g = (f, \geq)$ donnée sur un ensemble de solutions S, l'écart $f(x) - f(y)$ est une fonction-critère sur le produit cartésien $S \times S$. On notera $gg = (ff, \geq)$ cette fonction-critère, avec $ff(x, y) = f(x) - f(y)$. Le couple (g, gg) est alors appelé **graduation**.

Remarquons que rien n'empêche d'attribuer des propriétés additionnelles (vrai-critère, critère d'intervalle, etc.) à la fonction-critère gg, en accord avec l'information préférentielle à modéliser.

Maintenant, il est possible de classer les décisions selon qu'elles soient ou non influencées par ces informations intra-critères. Ces informations sont aussi appelées **axiomes de mesurabilité**. Ainsi, une décision influencée uniquement par l'ordre induit par la fonction-critère, est qualifiée d'**ordinale**. Prenons l'exemple du vrai-critère. D'abord, notons $\mathcal{U}(S, \quad)$ l'ensemble des fonctions de S dans . Dans ce cadre, $g = (f, \geq)$ étant donné, tout vrai-critère $g' = (f', \geq)$ tel qu'il existe une transformation strictement croissante ϕ de la variable réelle permettant d'obtenir f' à partir de f ($\Leftrightarrow f'(x) = \phi(f(x))$ pour tout $x \in S$), n'influence pas la décision ordinale sur des vrai-critères.

Le cas opposé, où la décision est influencée à la fois par l'ordre sur S et l'ordre sur $S \times S$, est qualifié de **cardinal**. Dans l'exemple de la graduation (g, gg) formée de deux vrai-critères, toute graduation de vrai-critères obtenue via une transformation affine positive, n'influence pas la décision cardinale sur des graduations de vrai-critères.

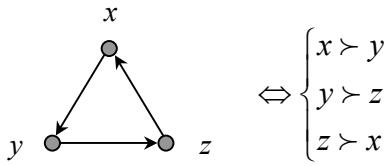
Les décisions peuvent aussi être classées selon d'autres axiomes de mesurabilité. Par exemple, l'**invariance par homothétie** (*ratio-scale mesurability*) : Dans le cas du vrai-critère, si f' est construite à partir d'un autre vrai-critère f en utilisant une valeur de référence, alors la décision n'est pas influencée.

Pour de plus amples informations, le lecteur est invité à consulter les références précitées. D'autres **axiomes d'invariance**, différents des axiomes de mesurabilité ont été utilisés lorsque la décision est influencé par plusieurs critères. Ce sont les axiomes de comparabilité. Nous y reviendrons lors de l'étude des profils à information interpersonnelle de type sympathie étendue (§ 4.1.3).

3.3.4. Les Relations Non-Binaires de Préférence

- Définition et interprétations.

L'axiome de cohérence des choix (exclusivement) binaires, a été aussi remis en question en TCS (voir SUZUMURA [1983], SEN [1986]). Le but est de se placer à un niveau d'abstraction plus élevé que le choix binaire, pour traiter certaines singularités du cas binaire. Prenons l'exemple d'un ensemble formé de trois solutions potentielles $S = \{x, y, z\}$ où les préférences s'expriment par une relation binaire de préférence $\succcurlyeq$ sur S formant un circuit de préférence stricte (Figure 13). Si un choix doit être fait, devons-nous conclure à l'absence d'une meilleure solution ? La seule connaissance de $\succcurlyeq$ n'est pas suffisante. Si nous voulons conclure, par exemple, à l'équivalence entre les trois solutions ou la préférence pour une solution, une possibilité est, d'accepter des choix initiaux sur des sous-ensembles non binaires de S.



$$\Leftrightarrow \begin{cases} x \succ y \\ y \succ z \\ z \succ x \end{cases}$$

Figure 13 : Exemple de circuit de préférences strictes.

Une **relation non-binaire de préférence** $rnp(S)$ sur un ensemble de solutions potentielles S est un couple $(\mathcal{A}, C)$ où :

- $\mathcal{A}$ est un sous-ensemble non vide de $\mathcal{P}(S) \setminus \{\emptyset\}$, et
- C une fonction associant à chaque élément A de $\mathcal{A}$ un de ses sous-ensembles non vide $C(A) : \emptyset \neq C(A) \subseteq A$. C est appelée **fonction de choix**[69], $C(A)$ est appelé **ensemble de choix** de A, et une solution $x \in C(A)$ est dite **solution choisie** à partir de A. La fonction de choix est dite **complète** lorsque $\mathcal{A} = \mathcal{P}(S) \setminus \{\emptyset\}$.

L'ensemble de choix $C(A)$ peut contenir plus d'une solution, et pour qu'un choix final d'une unique solution sur A, puisse se faire, il est nécessaire d'accepter l'équivalence entre les éléments de $C(A)$.

Concernant les possibilités de retour aux choix binaires sur S, plusieurs théories ont été développées dont celles des préférences révélées (voir par exemple SEN [1971], [1986 § 4], [1993], [1997*a*]).

De même que pour les choix binaires, il peut exister plusieurs niveaux d'appréciation de la qualité d'un choix sur un ensemble $A \subseteq S$. La notion de « brillant second » de ROY & BOUYSSOU [1993] en est un exemple.

Ainsi, considérons un ensemble partiellement ordonné $(NQ, \succcurlyeq_{NQ})$ de niveaux de qualité possibles admis sur nos choix. Alors à tout ensemble $A \subseteq S$ un niveau de qualité $\alpha \in (NQ, \succcurlyeq_{NQ})$ peut être attribué à son ensemble de choix $C(A)$. Nous noterons $C(A, \alpha)$. Quelques fois, à un ensemble A peut être associé plusieurs fonctions de choix de qualité différente, et vérifiant $C(A, \alpha_1) \subseteq C(A, \alpha_2)$ lorsque $\alpha_1 \succcurlyeq_{NQ} \alpha_2$. Cette propriété indique que plus la qualité de l'ensemble de choix augmente, plus l'ensemble des solutions choisies est restreint.
Remarquons, la ressemblance entre l'ensemble structuré des niveaux de qualité et la structure de préférence utilisée dans le cas binaire.

Bien qu'en théorie ce modèle offre des avantages indéniables pour la modélisation de préférences complexes, dans la pratique, il pose d'énormes problèmes de complexité[70]. Seules les relations non-binaires possédant des propriétés particulières de réduction de la complexité peuvent être envisagées. Cette théorie est néanmoins nécessaire pour étudier certains problèmes d'aide à la décision, comme nous le verrons aux § 5.3.4 et 5.3.5.

3.4. Aider à Choisir avec une Relation Binaire de Préférence

Rappelons que notre problématique décisionnelle est celle du choix d'une unique solution. Mais, nous pouvons résoudre notre problème en utilisant dans le processus de décision des problématiques techniques de choix, de consistance, ou encore de rangement ou de tri, etc. (§ 3.2.3). Dans cette section nous nous intéressons à la problématique d'aide au choix dans le cas particulier où les préférences du problème sont décrites par une relation binaire de préférence[71].

Considérons un ensemble de solutions S supposé non vide et une relation binaire de préférence $\succcurlyeq$ sur S. Ici, aucune hypothèse n'est faite sur la manière dont le couple $(S, \succcurlyeq)$ est donné : représentations explicite ou en compréhension (§ 2.1.1.1). Notons $C(S, \succcurlyeq)$ le sous-ensemble de solutions non vide à déterminer en fonction de la relation binaire de préférence

[69] Voir Peter FISHBURN [1973] – *The Theory of Social Choice* – Princeton, Princeton University Press.

[70] La représentation explicite de ce modèle est inenvisageable lorsque $\mathrm{card}(\mathcal{A})$ est exponentiel en la taille des autres informations. Par exemple, si S est donnée explicitement, et que $rnp(S)$ est complète : $\mathcal{A} = \mathcal{P}(S) \setminus \{\emptyset\}$. Alors $\mathrm{card}(\mathcal{A}) = 2^{\mathrm{card}(S)} - 1$.

[71] Néanmoins, pour des travaux portants directement sur les relations binaires de préférence valuées, se reporter à PERNY [1992], PERNY [1998]. Pour les préférences utilisant la logique tétravalente se référer à TSOUKIÀS [1999], et les relations non-binaires de préférences, voir SUZUMURA [1983], SEN [1986 § 4].

$\succcurlyeq$ sur S. Nous l'appelons **ensemble de choix**, et ses éléments **solutions choisies** ou **satisfaisantes** ou **de bonne qualité**.

Nous nous intéressons au problème de **caractérisation axiomatique de l'ensemble de choix d'une relation binaire de préférence**, apportant des réponses aux questions suivantes : Quel est le système d'axiomes de choix sur lequel repose le sous-ensemble $C(S, \succcurlyeq)$, en fonction des propriétés de la relation binaire $\succcurlyeq$? Comment peut-on relaxer la propriété d'optimalité pour continuer à identifier un sous-ensemble non vide satisfaisant ?

Après avoir présenté les axiomes les plus courants que sont l'optimalité, la maximalité (§ 3.4.1) et la stabilité maximale (§ 3.4.2) pour caractériser les ensembles de choix, nous introduisons d'autres systèmes axiomatiques (domination, noyau, fermeture transitive, etc.) de la littérature utilisés lorsque l'optimalité et la maximalité ne sont plus valables (§ 3.4.3). Un certain nombre de reproches peuvent être faits à ces caractérisations usuelles, et de nouveaux axiomes sont présentés pour palier à ces défauts. Tout d'abord, deux axiomes garantissant la qualité des solutions choisies (l'optimalité relative et la maximalité relative) sont introduits, généralisant l'optimalité et la maximalité (§ 3.4.4). Puis les axiomes d'indiscernabilité séparant la redondance et l'approximation sont présentés et comparés à l'existant (§ 3.4.5). Cette section est clôturée par une synthèse (§ 3.4.6).

3.4.1. Optimalité et Maximalité

En optimisation classique, le préordre complet défini par un vrai-critère, identifie les solutions satisfaisantes comme celles qui sont optimales. Rappelons la définition de l'optimalité dans le cadre plus général de la relation binaire de préférence.

L'**ensemble optimal** $B(S, \succcurlyeq)$ de S selon la relation binaire de préférence $\succcurlyeq$ est un sous-ensemble de S qui regroupe les éléments qui sont au moins aussi bons que tout élément de S :

$$\forall\, x \in B(S, \succcurlyeq) \text{ et } \forall\, y \in S,\ \ x \succcurlyeq y\ . \ (\textbf{axiome d'optimalité}) \qquad (3\text{–}5)$$

Un élément de l'ensemble optimal est appelé **solution optimale**. Nous dirons qu'un sous-ensemble A de S possède la propriété d'**optimalité**, si $A \neq \varnothing$ et $A \subseteq B(S, \succcurlyeq)$. L'ensemble optimal possède une propriété remarquable : toutes les solutions optimales sont indifférentes.
Cet ensemble existe pour des r.b.p. possédant des propriétés structurelles moins exigeantes que le préordre total. Par conséquent, lorsqu'il existe au moins une solution optimale pour une instance $(S, \succcurlyeq)$ donnée, les solutions choisies sont identifiées aux solutions optimales :

$$\text{Si}\ \ B(S, \succcurlyeq) \neq \varnothing\ \ \text{ alors }\ \ C(S, \succcurlyeq) = B(S, \succcurlyeq)\ .$$

Nous parlons d'**ensemble de choix optimisant** (traduit de "*optimizing choice set*" de SEN [1997*a*]), lorsque l'ensemble de choix s'appuie sur l'axiome d'optimalité.

Pour les cas où l'ensemble optimal est vide, une autre condition, la maximalité, a été adoptée unanimement dans la littérature. Bien que moins exigeante que l'optimalité, elle est jugée suffisante pour que x soit un élément de l'ensemble de choix.

L'**ensemble maximal** $M(S, \succcurlyeq)$ de S selon une relation binaire $\succcurlyeq$ est le sous-ensemble de S pour lequel il n'existe aucun élément dans S meilleur que ses éléments.

$$\forall\, x \in M(S, \succcurlyeq) \text{ et } \forall\, y \in S,\ \ \text{non}\,[y \succ x]\ . \ (\textbf{axiome de maximalité}) \qquad (3\text{–}6)$$

Un élément de cet ensemble est appelé **solution maximale**[72]. Nous dirons que l'ensemble maximal possède la propriété de **maximalité**. Un ensemble de choix satisfaisant la maxima-

[72] Elle est aussi appelée solution **S**-efficace par ROY & BOUYSSOU [1993 § 6.1.2 c)], **S** étant mis pour désigner la relation $\succcurlyeq$. Remarquons que la structure de préférence utilisée par ROY & BOUYSSOU [1993] est $\{\succcurlyeq, \preccurlyeq, \|\}$ et non pas *AF*(*RBP*). Nous renvoyons au § 3.3.2.1 pour quelques relations entre la relation de surclassement et la relation binaire de préférence.

lité est nommé ensemble de choix maximisant, et est automatiquement adopté lorsque l'ensemble optimal est vide :

$$\text{Lorsque } B(S, \succcurlyeq) = \varnothing \text{ et } M(S, \succcurlyeq) \neq \varnothing \text{ alors } C(S, \succcurlyeq) = M(S, \succcurlyeq) .$$

Contrairement à l'ensemble optimal, l'ensemble maximal est formé d'éléments soit incomparables, soit indifférents entre eux.

L'ensemble optimal et l'ensemble maximal sont reliés par le théorème suivant (SEN [1970], [1997*a*]) lorsque S est non vide et fini :

Théorème 1 : $B(S, \succcurlyeq) \subseteq M(S, \succcurlyeq)$, mais l'inverse est généralement faux. Les cas où $B(S, \succcurlyeq)$ et $M(S, \succcurlyeq)$ diffèrent peuvent être partitionnés en deux catégories :

(1) $B(S, \succcurlyeq)$ est vide tandis que $M(S, \succcurlyeq)$ ne l'est pas, et

(2) $B(S, \succcurlyeq)$ et $M(S, \succcurlyeq)$ sont tous les deux non vides, et $M(S, \succcurlyeq) \setminus B(S, \succcurlyeq)$ est aussi non vide.

Les preuves sont données dans les références précitées. Ce théorème confirme qu'une solution optimale est aussi maximale, mais une solution maximale peut ne pas être optimale. Le cas (1) informe que même lorsqu'il n'y a pas de solution optimale, il peut y avoir une solution maximale. Le cas (2) informe que l'ensemble maximal définit quelques fois un ensemble non vide plus grand que l'ensemble optimal. Par exemple l'ensemble $S = \{x, y, z\}$, et la relation binaire de préférence $\succcurlyeq$ qui, en plus de la réflexivité impose que $x \simeq y$ et $y \simeq z$, entraînent que $B(S, \succcurlyeq) = \{y\}$ et $M(S, \succcurlyeq) = S$. Remarquons que si $\succcurlyeq$ est transitive et si $B(S, \succcurlyeq) \neq \varnothing$, alors $B(S, \succcurlyeq) = M(S, \succcurlyeq)$.

Lorsque $\mathsf{card}(S) > 2$, la littérature en AMCD et TCS propose plusieurs conditions suffisantes sur la relation binaire $\succcurlyeq$, pour que l'ensemble de choix ait la propriété d'optimalité ou de maximalité : Le cas standard est la condition de transitivité de $\succcurlyeq$. Lorsque la relation $\succcurlyeq$ est un préordre total (resp. partiel), l'ensemble $B(S, \succcurlyeq)$ (resp. $M(S, \succcurlyeq)$) est non vide. SEN [1970] et [1997*a*] donne d'autres conditions suffisantes pour que ces sous-ensembles remarquables soient non vides. Notamment, il suffit que $\succcurlyeq$ soit P-acyclique (cf. § 3.3.1.2) et complète (resp. strictement P-acyclique) pour que $B(S, \succcurlyeq)$ (resp. $M(S, \succcurlyeq)$) soit non vide. A l'opposé donc, s'il existe au moins un circuit de préférence stricte de longueur supérieur ou égale à 3, nous ne sommes pas sûrs de l'existence d'une solution optimale (resp. maximale).

Remarquons que $M(S, \succcurlyeq)$ peut être un singleton ($\mathsf{card}(M(S, \succcurlyeq)) = 1$) même lorsque $B(S, \succcurlyeq) = \varnothing$ (c'est le cas de l'exemple (*a*) de la Figure 15).

3.4.2. Redondance et Approximation Exprimées par la Stabilité Maximale

Dans cette sous-section, nous nous intéressons à la question suivante : Un problème d'aide au choix doit-il toujours retourner toutes les solutions des ensembles optimaux ou maximaux, pour remplir sa fonction d'aide à la décision ? Dans le cadre des problèmes d'optimisation simple, il n'y a aucun signe préférentiel permettant de distinguer les éléments de l'ensemble optimal : ils ont même valeur. Par conséquent on peut se permettre de retourner une unique solution optimale. Dans un cadre plus général, ce n'est pas toujours le cas. Par exemple, dans le cadre de l'ensemble de choix maximisant, l'incomparabilité entre deux solutions maximales implique que ces solutions ont des propriétés qui leurs sont propres. N'en garder qu'une impliquerait un choix arbitraire, une perte d'information, un biais de la procédure de choix préjudiciable à une aide à la décision. Dans les écrits de l'AMCD et la théorie des graphes, la suppression de l'information redondante a généralement pris la forme de la propriété de stabilité.

Un sous-ensemble A d'un ensemble S est un **stable**[73] (ou stabilité interne ROY [1969, p. 192], [1970], ou indépendance) selon la relation binaire $\succcurlyeq$ si et seulement si

$$\forall\ (x, y) \in A \text{ et } x \neq y,\ x \parallel y\ .\ (\textbf{stabilité}) \qquad (3\text{–}7)$$

En d'autres termes le sous-ensemble structuré de $(S, \succcurlyeq)$ induit par A ne contient que des couples incomparables si et seulement si A est un stable. Nous dirons qu'un stable possède la propriété de **stabilité**. Cette propriété traduit une notion d'« assujettissement minimale à polémique » (extrait de ROY [1969-1970]). Nous étudierons au § 3.4.5 les pertes occasionnées par l'adoption d'un tel axiome pour caractériser une solution d'un problème de choix basé sur un couple $(S, \succcurlyeq)$.

La non redondance et l'approximation dans une problème de choix ont alors pris la forme de la stabilité maximale. Par conséquent tout algorithme retournant au plus une solution optimale (resp. maximale) par couple de solutions indifférentes optimales (resp. maximales) est jugé remplir son rôle d'aide à la décision. On note alors $SC(S, \succcurlyeq)$ un tel sous-ensemble de $C(S, \succcurlyeq)$ et on l'appelle **ensemble de choix stable maximal**. Sauf si $SC(S, \succcurlyeq) = C(S, \succcurlyeq)$ il existe plusieurs ensembles de choix stables maximaux pour un couple $(S, \succcurlyeq)$.

Un **ensemble de choix optimisant stable maximal** $SB(S, \succcurlyeq)$ est toujours formé d'une seule solution optimale : $\mathrm{card}(SB(S, \succcurlyeq)) = 1$, car les solutions optimales sont deux à deux indifférentes. Lorsque $B(S, \succcurlyeq) = \varnothing$ et que $M(S, \succcurlyeq) \neq \varnothing$, deux **ensembles de choix maximisants stables maximaux** $SM(S, \succcurlyeq)$ n'ont pas forcément même cardinal. Prenons le cas de $S = \{w, x, y, z, t\}$, et la relation binaire de préférence $\succcurlyeq$ (graphe (a) de la Figure 14) qui, en plus de la réflexivité impose que $w \simeq x$, $x \simeq y$, $y \simeq z$ et $y \simeq t$, lequel entraîne que $SM(S, \succcurlyeq) \in \{\ \{w, y\}, \{x, z, t\}, \{w, z, t\}\ \}$. Remarquons que la stabilité maximale n'implique pas le choix exactement d'une solution par couple de solutions indifférentes (c'est le cas de $\{w, z, t\}$). Ces résultats sont bien connus en théorie des graphes (BERGE [1983]). Un tel stable maximal est appelé **stable parfait**, et de la même manière des ensembles de choix stables parfaits peuvent être envisagés.

Remarque : les arcs réflexifs ont été omis.

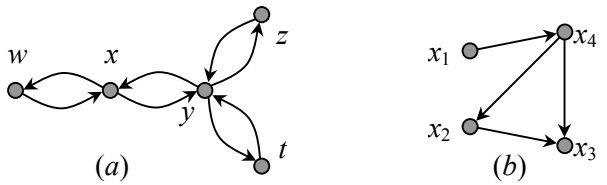


Figure 14 : Quelques exemples de graphes servant de support aux propriétés présentées.

Mais comment choisir des solutions satisfaisantes, lorsque l'ensemble maximal – et donc l'ensemble optimal – est vide ?

3.4.3. Au-Delà de l'Optimalité et de la Maximalité

Lorsqu'il n'existe pas de solution maximale pour un couple $(S, \succcurlyeq)$, qu'est il possible de faire ? Ne peut-on pas néanmoins identifier des solutions visiblement de meilleure qualité que d'autres ? Le jugement des solutions de meilleure qualité repose sur les axiomes de choix. Mais lorsque ceux-ci sont inconsistants (= identifient des sous-ensembles vides) pour une instance donnée, il est nécessaire d'en assouplir (remettre en question) certains. Cette manière de procéder (en relaxant au fur et à mesure une propriété) est l'une des manières d'investigation axiomatique d'un problème.

Présentons donc quelques axiomes de choix de la littérature. Des résultats similaires, voire équivalents, mais souvent découverts de façon indépendante, peuvent être trouvés à la fois en

[73] Cette notion est généralement définie pour les graphes orientés sans boucle. Elle a été adaptée aux graphes avec boucle en ajoutant la condition $x \neq y$.

théorie des graphes d'une part et en théorie du choix social et théorie des jeux d'autre part, du fait qu'une relation binaire de préférence est équivalente à un graphe orienté simple réflexif.

3.4.3.1. Domination et Absorption dans les Graphes Orientés

VON NEUMANN & MORGENSTERN [1947] ont proposé différents axiomes de substitution : la domination et l'absorption (voir aussi HENNING [1996], HAYNES *et al.* [1998*a*] et [1998*b*]). Nous les présentons ci-dessous :

Un sous-ensemble A d'un ensemble S est un **ensemble dominant**[74] selon une relation binaire $\succcurlyeq$ ssi

$$\forall\, y \in S \setminus A, \exists\, x \in A \text{ tel que } x \succcurlyeq y \text{ . (\textbf{domination})} \qquad (3\text{–}8)$$

Nous dirons qu'un ensemble dominant possède la propriété de **domination** (appelé stabilité externe par ROY [1969, p. 192], [1970]). Un ensemble dominant A est un **ensemble dominant minimal** pour $(S, \succcurlyeq)$ ssi il n'existe pas d'ensemble dominant pour $(S, \succcurlyeq)$ strictement inclus dans A.

Remarquons que la maximalité n'implique pas la domination ! En d'autres termes, un ensemble maximal n'est pas forcément un ensemble dominant. La Figure 15 nous montre l'exemple d'un couple $(S, \succcurlyeq)$ où il existe deux ensembles dominants minimaux pour l'inclusion : $\{x_1, x_3\}$ et $\{x_1, x_4\}$; alors que l'ensemble maximal $\{x_1\}$ est strictement inclus dans les deux premiers. Par contre, toute solution optimale forme à elle seule un ensemble dominant minimal, et tout ensemble maximal est contenu dans un ensemble dominant minimal.

De même, un sous-ensemble A d'un ensemble S est un **ensemble absorbant** selon une relation binaire $\succcurlyeq$ ssi

$$\forall\, y \in S \setminus A, \exists\, x \in A \text{ tel que } y \succcurlyeq x \text{ . (\textbf{absorption})} \qquad (3\text{–}9)$$

3.4.3.2. Le Noyau et ses Relaxations

La notion de noyau (*kernel*) est apparue avec les travaux de VON NEUMANN & MORGENSTERN [1944, 1947, chap. XII] en théorie de jeux, sous le nom de "solution". Dès lors, les noyaux ont été traités en théorie de graphes (ROY [1970], BERGE [1983, chap. 14], parmi d'autres. Voir GHOSHAL *et al*. [1998] pour une synthèse)[75].

Nous appelons **noyau** $N(S, \succcurlyeq)$ de S selon $\succcurlyeq$, un ensemble qui vérifie conjointement l'axiome de domination et l'axiome de stabilité. Par exemple, l'exemple (*b*) de la Figure 14, a pour noyau : $\{x_1, x_2\}$. Il existe des couples $(S, \succcurlyeq)$ sans noyau comme l'exemple (*b*) de la Figure 15, ou avec plusieurs noyaux comme l'exemple (*a*) de la Figure 17. Le terme de noyau est aussi utilisé pour désigner un ensemble de choix satisfaisant la domination et la stabilité.

Lorsque la relation binaire de préférence n'est pas transitive, la notion de noyau est souvent employée pour identifier un ensemble de choix (VON NEUMANN & MORGENSTERN [1944], ROY & BOUYSSOU [1993], parmi d'autres), alternativement à celle d'ensemble de choix maximisant stable maximal. Quels sont les liens entre les systèmes axiomatiques sous-jacents à ces deux types d'ensembles ?

---

[74] ROY & BOUYSSOU [1993, § 6.1.2.] parlent d'**ensemble surclassant** dans le cadre de la relation de surclassement (§3.3.2.1).

[75] En théorie des graphes, les notions de noyau et de solution d'un graphe sont des concepts duaux, le premier étant un stable absorbant, et le second un stable dominant (GHOSHAL *et al*. [1998]). Ces 2 notions ont quelques fois été employées l'une pour l'autre.

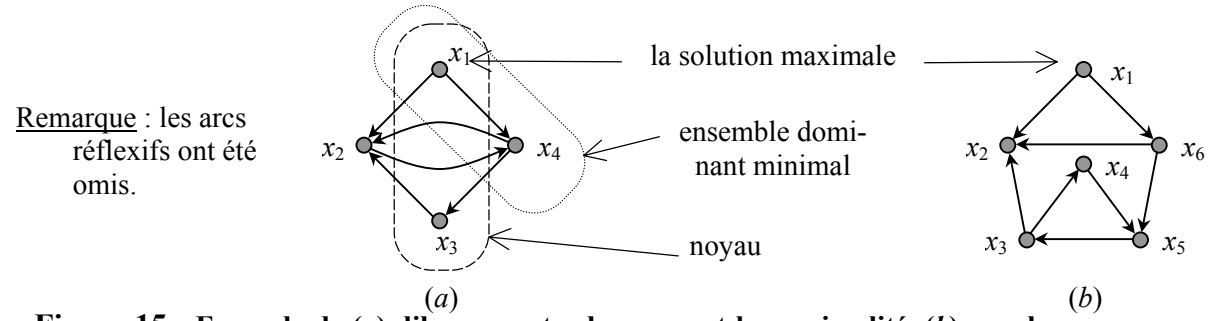


Figure 15 : Exemple de (*a*) dilemme entre le noyau et la maximalité, (*b*) graphe sans noyau et avec un ensemble maximal.

Un noyau se présente comme un ensemble d'éléments deux à deux non comparables, qui, pris globalement, est au moins aussi bon que tout autre élément situé hors du noyau (ROY [1970 p. 33]). Mais rien n'empêche qu'un élément de S situé hors du noyau soit au moins aussi bon que (voire préféré à) au moins un élément du noyau (certains éléments hors du noyau peuvent être préférés à un ou plusieurs éléments du noyau), dès que celui-ci est à son tour dominé par un autre élément du noyau. Nous avons pour exemple le graphe (*a*) de la Figure 14 : l'élément x_3 du noyau $\{x_1, x_3\}$ est moins bon que l'élément hors du noyau x_4 qui est lui dominé par x_1. Ce type de singularité peut être jugé inacceptable pour un ensemble de choix. D'un autre coté, la maximalité tolère des solutions non choisies, qui sont incomparables avec toutes les solutions choisies (par exemple, l'unique solution maximale x_1 de l'exemple (*b*) de la figure précitée, est incomparable à la solution x_3 et aucune solution maximale ne lui est préférée). Ce qui peut ne pas être non plus acceptable. La conjonction de ces trois axiomes permet d'y palier. Un ensemble de choix satisfaisant à la fois la maximalité, la domination et la stabilité peut être utilisé comme compromis lorsque l'ensemble optimal est vide. Mais ce type d'ensemble de choix n'est valable que dans une sphère plus restreinte que celle de la maximalité et du noyau.

Remarquons aussi que l'existence d'un ensemble maximal n'implique pas toujours celle d'un noyau (exemple (*b*) de la Figure 15). De même, l'existence d'un noyau n'implique pas celle d'un ensemble maximal (Par exemple, tout graphe formé uniquement d'un circuit de longueur paire). Lorsque la relation binaire de préférence $\succcurlyeq$ est transitive, alors tout ensemble de choix maximisant stable maximal est aussi un noyau ; et vice versa (la preuve est donnée en Annexe B1, Proposition 28). De plus, tout ensemble de choix optimisant stable maximal est un noyau ; mais la réciproque est fausse. La Figure 16 récapitule schématiquement les différentes relations existant entre le noyau, l'optimalité et la maximalité.

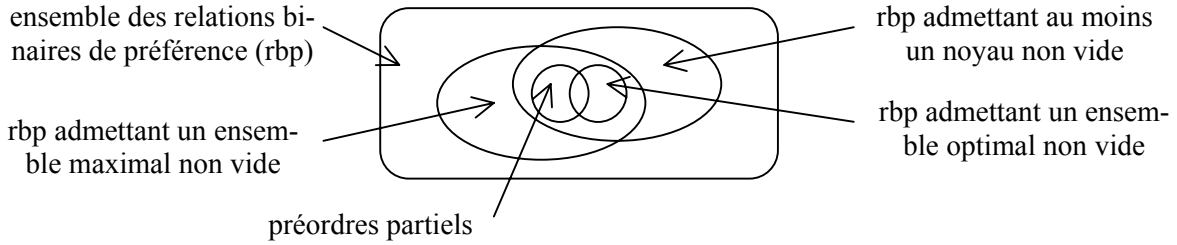


Figure 16 : Schéma récapitulant les relations entre la maximalité, l'optimalité, le noyau et la transitivité.

Sachant qu'il existe des graphes réflexifs sans noyau, plusieurs travaux ont porté sur la relaxation des axiomes de domination et stabilité. Ainsi les notions de quasi-noyau (VINCKE [1977]), semi-noyau et (k, l)-noyau (GHOSHAL *et al*. [1998] pour une synthèse) sont apparues. Nous appelons collectivement ces ensembles, les noyaux généralisés. La stabilité est remplacée par l'existence, pour tout couple d'éléments d'un noyau généralisé, d'un plus court che-

min de longueur minimale spécifiée ≥ 2. La domination est remplacée par l'existence, pour tout élément y hors du noyau généralisé, d'un plus court chemin de longueur maximale spécifiée, partant d'un élément du noyau généralisé vers y. Les noyaux généralisés existent pour tout couple $(S, \succcurlyeq)$. Nous ne nous étendons pas sur le sujet ici et renvoyons aux références précitées. Un certain nombre de comparaisons reste à réaliser entre les noyaux généralisés et l'ensemble maximal.

3.4.3.3. Fermeture Transitive, Base

La fermeture transitive a été aussi très utilisée (ROY [1970], SEN [1986], SUZUMURA [1983] parmi d'autres). Étant donné un ensemble non vide S structuré par une relation binaire de préférence $\succcurlyeq$ et une attitude $\alpha \in \mathcal{PR}(AF(RBP))$, la **fermeture transitive de** $(S, \succcurlyeq)$ **selon** α est la relation binaire de préférence $\succcurlyeq_{\tau(\succcurlyeq, \alpha)}$ définie sur S de la manière suivante :

$$\text{Pour } x, y \in S \text{ et } x \neq y, \quad x \succcurlyeq_{\tau(\succcurlyeq, \alpha)} y \Leftrightarrow \begin{cases} \exists \text{ une séquence } z_1 \alpha z_2, \cdots, z_{k-1} \alpha z_k \\ \text{avec } z_1 = x \text{ et } z_k = y \end{cases}.$$

Par construction, les fermetures transitives $\succcurlyeq_{\tau(\succcurlyeq, \alpha)}$ sont des relations binaires de préférence transitives. Par conséquent, leur ensemble maximal existe, pour tout $(S, \succcurlyeq)$ avec S non vide. Deux variantes sont apparues en TCS (SEN [1986, § 4.1]) : la **fermeture transitive faible** $\succcurlyeq_{\tau(\succcurlyeq, \succcurlyeq)}$ et la **fermeture transitive forte** $\succcurlyeq_{\tau(\succcurlyeq, \succ)}$. L'ensemble maximal $M(S, \succcurlyeq_{\tau(\succcurlyeq, \alpha)})$ de $\succcurlyeq_{\tau(\succcurlyeq, \alpha)}$ peut faire office d'ensemble de choix pour la relation $\succcurlyeq$, et est noté τ_α-$M(S, \succcurlyeq)$. En AMCD (ROY [1970, p. 37]), la notion de **base** $\tau_\succcurlyeq$-$SM(S, \succcurlyeq)$ de $(S, \succcurlyeq)$ est apparue comme étant un ensemble de choix maximisant stable maximal pour $(S, \succcurlyeq_{\tau(\succcurlyeq, \succcurlyeq)})$.

Illustrons ces ensembles avec l'exemple de la Figure 15 : La fermeture transitive faible du graphe est formée de deux classes d'équivalences : $\{x_1\}$ et $\{x_2, x_3, x_4\}$, et l'unique base est égale à l'ensemble maximal. La fermeture transitive forte du graphe aboutit à l'ordre total suivant : $x_1 \succ x_4 \succ x_3 \succ x_2$. L'ensemble de choix associé à la fermeture transitive forte est donc aussi égal à l'ensemble maximal. Remarquons que les fermetures transitives faible et forte peuvent aboutir à des ensembles de choix différents de l'ensemble optimal ou maximal. Prenons l'exemple (*b*) de la Figure 17, où l'ensemble optimal est $\{x_6\}$, la base est S et où $\tau_\succ$-$SM(S, \succcurlyeq) \in \{ \{x_6, x_i\}$, avec $1 \leq i < 6\}$. En général, $\tau_\succ$-$M(S, \succcurlyeq) \subseteq \tau_\succcurlyeq$-$M(S, \succcurlyeq)$. Lorsque l'ensemble optimal est non vide, $B(S, \succcurlyeq) \subseteq \tau_\succ$-$M(S, \succcurlyeq)$, et lorsque l'ensemble maximal est non vide $M(S, \succcurlyeq) \subseteq \tau_\succ$-$M(S, \succcurlyeq)$. Pour d'autres exemples et références voir SEN [1986].

Ces différents axiomes permettent de caractériser des ensembles de choix qui conduisent à des impasses pour les concepts usuels d'optimalité et de maximalité.

3.4.4. Choisir avec Prudence : Optimalité Relative et Maximalité Relative

3.4.4.1. Optimalité et Maximalité Relatives : Définition et Caractérisations

De nombreux reproches peuvent être faits à l'encontre des axiomes véhiculés par les notions de noyau et de base, pour l'identification de solutions satisfaisantes. Le noyau n'existe pas toujours, et bien qu'il semble adapté à certains problèmes de stratégie (BERGE [1983, chap. 14]) et de théorie des jeux (HAYNES et *al.* [1998*a*] et [1998*b*]), il peut être une faible consolation pour caractériser l'ensemble des solutions intuitivement[76] satisfaisantes. En effet, l'adoption du noyau implique qu'il est plus important d'avoir des solutions choisies incompa-

[76] Remarquons que seule l'intuition permet de nous guider. Cette intuition est grandement stimulée par les exemples construits. L'intuition peut, certes, nous induire en erreur. Mais les exemples d'école construits sont très simples, et généralement servent de base à des instances de plus grande taille où l'intuition a plus de chance de se tromper.

rables que des solutions choisies pouvant être pires que des solutions non choisies. La notion de base permet de palier à cet inconvénient.

La base, elle, fait une approximation grossière de l'ensemble des solutions à choisir. Prenons l'exemple (*b*) de la Figure 17, où l'ensemble optimal est $\{x_6\}$ alors que la base est tout S. La base ne garantit pas que l'information retournée sera minimale.

La définition de la problématique d'aide au choix identifie deux types d'axiomes de choix principaux. Les axiomes garantissant la qualité des solutions choisies pour les protagonistes, et les axiomes exprimant que l'information retournée doit être minimale en nombre d'éléments. Dans cette section, nous présentons deux nouveaux axiomes de choix mettant en application ces principes dans le cadre d'une relation binaire de préférence.

Un sous-ensemble A d'un ensemble S est un **ensemble relativement optimal** (resp. un **ensemble relativement maximal**), et on note e.r.o. (resp. e.r.m.), selon une relation binaire de préférence $\succcurlyeq$ ssi :

$$\forall\, x \in A, \text{ et } \forall\, y \in S \setminus A,\ x \succcurlyeq y\ . \text{ (\textbf{optimalité relative})} \tag{3–10}$$

$$\forall\, x \in A, \forall\, y \in S \setminus A,\ \text{non}\,[y \succ x]\ . \text{ (\textbf{maximalité relative})} \tag{3–11}$$

Nous nous sommes informés sur les propriétés et classes de graphes les plus courantes (BERGE [1983], BRANDSTÄDT *et al.* [1999]), sans trouver d'étude sur de tels axiomes. Il en est de même en TCS et en AMCD. Nous avons donc entrepris une succincte étude. Ces axiomes vont dans la continuité de l'optimalité et la maximalité, comme nous allons le voir par la suite.

Remarque : en terme d'écriture des définitions, la maximalité relative ne se différencie de la maximalité que par l'ensemble de définition de y, lequel est $S \setminus A$ pour le premier et S tout entier pour le second. Cette petite variation dans l'écriture induit à l'opposé une énorme différence dans l'identification qu'elle fait. Il en est de même de l'optimalité relative par rapport à l'optimalité. Voici quelques propriétés reliant l'optimalité relative à la maximalité relative :

Propriété 1 : Tout e.r.o. est un e.r.m.. Il existe toujours un e.r.o. et un e.r.m., sauf si S est vide, car S est lui-même un e.r.o. et donc un e.r.m..

La preuve est immédiate. Intéressons-nous aux ensembles relativement optimaux (resp. maximaux) et minimaux pour l'inclusion. Voici leur définition :

Un **ensemble relativement optimal** *MRO* (resp. **maximal** *MRM*) est **minimal pour l'inclusion**, et on note e.r.o.m.i. (resp. e.r.m.m.i.) s'il n'existe aucun ensemble relativement optimal (resp. maximal) strictement inclus dans *MRO* (resp. *MRM*).

Pour un S donné, ces ensembles ne sont pas toujours uniques. L'exemple (*a*) de la Figure 17 permet d'illustrer ce propos en montrant un couple $(S, \succcurlyeq)$ comportant deux e.r.o.m.i. $\{x_1, x_3\}$ et $\{x_2, x_4\}$. L'exemple (*b*) montre un couple $(S, \succcurlyeq)$ ayant deux e.r.o.m.i. $\{x_6\}$ et $\{x_1, \ldots, x_5\}$, illustrant bien que, la taille n'est pas un invariant chez les e.r.o.m.i. d'un couple $(S, \succcurlyeq)$.

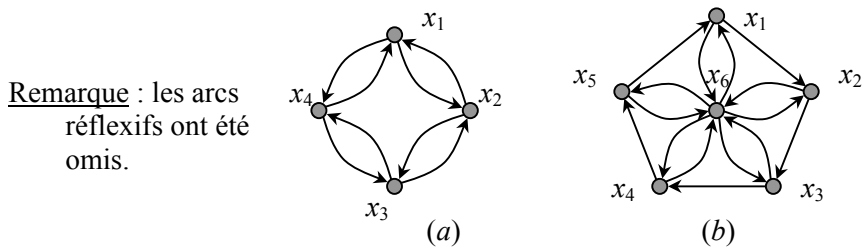


Figure 17 : Exemples de graphes illustrant les ensembles relativement optimaux et minimaux pour l'inclusion.

Ces deux axiomes (optimalité relative et maximalité relative) sont qualifiés de prudents, dans le sens où les solutions choisies par les e.r.o.m.i. et e.r.m.m.i. sont toutes celles qui auraient pu être dans un ensemble optimal ou maximal, s'il n'y avait pas d'intransitivités au

niveau des solutions de bonne qualité. Ils identifient un ensemble de solutions qui ont un comportement correspondant à l'optimalité (resp. la maximalité) avec l'extérieur, mais qui entre eux sont contradictoires. Les deux propositions suivantes permettent de confirmer ces affirmations.

La première proposition décrit d'autres propriétés caractéristiques des ensembles relativement optimaux et minimaux pour l'inclusion (e.r.o.m.i.).

Proposition 2 : Étant donnés deux e.r.o.m.i. différents MRO_1 et MRO_2 d'un ensemble non vide S structuré par une relation binaire de préférence $\succcurlyeq$, alors

(a) Deux solutions qui ne font pas partie du même e.r.o.m.i. sont indifférentes :

$$\forall\ (x, y) \in (MRO_1 \setminus MRO_2) \times (MRO_2 \setminus MRO_1),\quad x \simeq y$$

(b) Leur intersection est vide : $MRO_1 \cap MRO_2 = \varnothing$

Notons MRO, l'un des deux e.r.o.m.i..

(c) Si $\mathrm{card}(MRO) \geq 2$ alors $\forall$ solution $x \in MRO$, $\exists\ y \in MRO$ telle que $\mathrm{non}[x \simeq y]$

(d) Si $\mathrm{card}(MRO) \geq 3$ et que pour une solution $x \in MRO$, $\exists\ y \in MRO$ telle que $x \succ y$ (resp. $x \prec y$), alors $\exists\ z \in MRO \setminus \{x, y\}$ telle que $\mathrm{non}[x \succcurlyeq z]$ (resp. $\mathrm{non}[z \succcurlyeq x]$)

(e) Si S est fini, alors $\forall\ x, y \in MRO$, soit $x = y$, soit $\exists\ x_1, \ldots, x_{r-1} \in MRO$ avec $r \geq 1$, tel que $\mathrm{non}[x \succcurlyeq x_1]$, $\mathrm{non}[x_i \succcurlyeq x_{i+1}]$ pour tout $1 < i < r$, et $x_r = y$.

Preuve : Tout d'abord remarquons que $MRO_1 \setminus MRO_2 \neq \varnothing$ et $MRO_2 \setminus MRO_1 \neq \varnothing$ ($\Leftrightarrow$ l'un ne peut pas être inclus dans l'autre), car MRO_1 et MRO_2 sont différents et sont des e.r.o.m.i. (minimaux par inclusion), donc ne peuvent pas contenir un ensemble relativement optimal. La démonstration de chaque assertion est donnée en Annexe B1.

Les assertions (c) à (e) de la proposition précédente permettent de caractériser un e.r.o.m.i. :

Proposition 3 : Un e.r.o. MRO est un e.r.o.m.i. ssi (a) il est réduit à un élément, (b) il est formé de deux éléments incomparables, (c) il est de cardinal supérieur à 2 et vérifie les assertions (c) à (e) de la Proposition 2.

Preuve : Voir Annexe B1.

Une proposition équivalente est valable pour les e.r.m.m.i. :

Proposition 4 : Étant donné deux e.r.m.m.i. différents MRM_1 et MRM_2 d'un ensemble non vide S structuré par une relation binaire de préférence $\succcurlyeq$, alors

a) Deux solutions qui ne font pas partie du même e.r.m.m.i. sont indifférentes ou incomparables :

$$\forall\ (x, y) \in (MRM_1 \setminus MRM_2) \times (MRM_2 \setminus MRM_1),\quad x \simeq y \text{ ou } x \parallel y$$

b) Leur intersection est vide : $MRM_1 \cap MRM_2 = \varnothing$

Notons MRM, l'un des deux e.r.m.m.i..

c) Si $\mathrm{card}(MRM) \geq 2$ alors $\forall$ solution $x \in MRM$, $\exists\ y, z \in MRM \setminus \{x\}$ telles que $y \neq z$, $x \succ y$ et $x \prec z$

d) MRM a un cardinal toujours différent de 2.

e) Si S est fini et si $\mathrm{card}(MRM) > 2$, alors il existe un circuit dans le sous-graphe partiel[77] de $(S, \succcurlyeq)$ induit par MRM et $\succ$, contenant tous les sommets de MRM

Preuve : Voir Annexe B1.

3.4.4.2. Ensembles Relatifs Minimaux : Comparaison

[77] Nous appelons **sous-graphe partiel** d'un graphe $G = (U, A)$ induit par U' et A', avec $U' \subset U$ et $A' \subset A$ le graphe $(U', A' \cap A_{U'})$, avec $A_{U'} = \{(x, y) \in A$ tel que : $x \in U'$ et $y \in U'\}$.

Comparons maintenant les e.r.o.m.i. et les e.r.m.m.i.. La proposition suivante résume quelques propriétés :

Proposition 5 : Étant donné un couple $(S, \succcurlyeq)$ formant un ensemble non vide structuré par une relation binaire de préférence, alors

(a) Tout e.r.o.m.i. contient un e.r.m.m.i.
(b) Si un e.r.o.m.i. *MRO* est un singleton, alors *MRO* est un e.r.m.m.i.
(c) Si $\mathsf{card}(S)$ est fini, et si le sous-graphe partiel de $(S, \succcurlyeq)$ induit par un e.r.o.m.i. *MRO* et $\succ$, est fortement connexe, alors *MRO* est un e.r.m.m.i.

Preuve : L'assertion (a) vient du fait que tout e.r.o. – et donc tout e.r.o.m.i. – est un e.r.m., et que tout e.r.m. contient un e.r.m.m.i.. L'assertion (c) est directement déduite de l'assertion (a). (d) est une conséquence directe de l'assertion (e) de la Proposition 4 car dire qu'un graphe est fortement connexe, c'est dire qu'il existe un circuit simple contenant tous ses sommets. ❑

Proposition 6 : Supposons que S soit non vide et fini. L'algorithme suivant permet, à partir d'un e.r.m.m.i. *MRM*, de construire un e.r.o.m.i. *MRO* tel que $MRM \subseteq MRO$.

soit $MRO \leftarrow MRM$; $A \leftarrow MRM$.
tant que ($A \neq \varnothing$) **faire**
  <u>choisir</u> $x \in A$
  $\Gamma \leftarrow \{y \in S \setminus MRO$ tel que : $\mathsf{non}[x \succcurlyeq y]\}$
  $MRO \leftarrow MRO \cup \Gamma$
  $A \leftarrow A \cup \Gamma \setminus \{x\}$
fin tant que

Preuve : L'inclusion $MRM \subseteq MRO$ est acquise car l'algorithme ne fait qu'ajouter des éléments de S à *MRM*. Montrons maintenant que *MRO* est un e.r.o. : L'algorithme inclus dans *MRO* tout élément de S qui, à partir de *MRM*, est accessible de proche en proche par la relation $\mathsf{non}[x \succcurlyeq y]$. Il s'arrête lorsqu'il n'existe plus de y dans $S \setminus MRO$ tel que pour tout $x \in MRO$, $x \parallel y$ ou $x \prec y$. C'est-à-dire que $\forall\, y \in S \setminus MRO$ et $\forall\, x \in MRO$, $x \succcurlyeq y$. Ce qui est équivalent à écrire que *MRO* est un e.r.o.. *MRO* est aussi un e.r.o.m.i. car il vérifie la Proposition 3. En effet, tout e.r.m.m.i. *MRM* vérifie la propriété (e) de la Proposition 2 vérifiée par tout e.r.o.m.i.. De plus, tout élément ajouté y dans *MRM* pour obtenir *MRO* vérifie $\mathsf{non}[x \succcurlyeq y]$ pour tout $x \in MRM$. Ce qui implique les assertions (c) à (e) de la Proposition 2, et par conséquent la Proposition 3. Ainsi, *MRO* est un e.r.o.m.i.. ❑

Une des conséquences de cet algorithme est que, si S est fini, alors tout e.r.m.m.i. est contenu dans un e.r.o.m.i..

3.4.4.3. Optimalité Relative et Optimalité : Comparaison

Étudions maintenant la relation existant entre l'ensemble optimal et les e.r.o.m.i.. Voici quelques propriétés :

Proposition 7 : Étant donnée une relation binaire de préférence sur un ensemble non vide $(S, \succcurlyeq)$, alors

(a) S'il existe un e.r.o.m.i. de cardinal 1 alors l'ensemble optimal $B(S, \succcurlyeq)$ est égal à l'union des e.r.o.m.i. de cardinal 1
(b) S'il n'existe pas d'e.r.o.m.i. de cardinal 1 alors $B(S, \succcurlyeq) = \varnothing$
(c) Si l'on accepte l'axiome d'optimalité et de stabilité pour caractériser l'ensemble des solutions à choisir, alors tout e.r.o.m.i. de cardinal 1 est un ensemble de choix optimisant stable maximal

Preuve : Ces assertions découlent directement de la définition d'un ensemble optimal et d'un ensemble de choix optimisant stable. ❑

L'exemple (*b*) de la Figure 17 possède 2 e.r.o.m.i. dont l'un est de cardinal 1, et définit l'ensemble optimal. S'il n'existe pas d'e.r.o.m.i. de cardinal 1, tout e.r.o.m.i. de cardinalité minimum peut faire office d'ensemble de choix. Un ensemble de choix basé sur la notion d'e.r.o.m.i. est appelé **ensemble de choix relativement optimisant**. Un tel ensemble existe pour tout ensemble non vide S structuré par n'importe quelle relation binaire de préférence $\succcurlyeq$.

3.4.4.4. Maximalité Relative et Maximalité : Comparaison

Une proposition équivalente à la Proposition 7 est valable pour les e.r.m.m.i.. Il suffit de remplacer le terme e.r.o.m.i. par e.r.m.m.i., et ensemble optimal $B(S, \succcurlyeq)$ par ensemble maximal $M(S, \succcurlyeq)$ pour les assertions (a) et (b). L'assertion (c) par contre est différente et prend la forme de la proposition suivante :

Proposition 8 : Étant donnée une relation binaire de préférence sur un ensemble non vide $(S, \succcurlyeq)$, si S est fini et que les e.r.m.m.i. de cardinal 1 de $(S, \succcurlyeq)$ sont donnés en extension, alors il existe une procédure simple (algorithme glouton) permettant de construire n'importe quel ensemble maximisant stable maximal.

Preuve : L'algorithme est le suivant : Supposons que A contienne initialement les solutions de S dont chaque élément est l'unique élément d'un e.r.m.m.i. de cardinal 1. Si A est non vide, choisir un élément de A, le supprimer de A, et supprimer de A tous les éléments qui lui sont indifférents. Réitérez jusqu'à ce que A soit vide. Les solutions choisies forment un ensemble maximal qui de plus est stable maximal par construction. ❑

Cette proposition peut être généralisée au cas où on posséderait simplement un algorithme retournant, tant qu'il en existe, un e.r.m.m.i. de cardinal 1 qui ne soit pas inclus dans un ensemble donné, et qui soit incomparable à tous les éléments de cet ensemble. Selon la taille d'un ensemble de choix maximisant stable maximal par rapport à la taille du codage de $(S, \succcurlyeq)$, l'algorithme est sortie-polynomial (fin du § 2.1.1.2). Nous ne le présentons pas ici.

Remarquons que l'union des e.r.m.m.i. de cardinal 1 contient l'union des e.r.o.m.i. de cardinal 1. En effet, si nous supposons qu'il existe pour un couple $(S, \succcurlyeq)$ des e.r.o.m.i. et des e.r.m..m.i. de cardinal 1, alors nous avons : L'union des e.r.m.m.i. de cardinal 1 est égale à l'ensemble maximal. Ce dernier contient l'ensemble optimal (Théorème 1) qui est égal à l'union des e.r.o.m.i. de cardinal 1.

S'il n'existe pas d'e.r.m.m.i. de cardinal 1, tout e.r.m.m.i. de cardinalité minimum peut faire office d'ensemble de choix. Un ensemble de choix basé sur la notion d'e.r.m.m.i. est appelé **ensemble de choix relativement maximisant**. Un tel ensemble existe pour tout ensemble non vide S structuré par n'importe quelle relation binaire de préférence $\succcurlyeq$. Remarquons que, contrairement à l'optimalité relative, un ensemble de choix basé uniquement sur la maximalité relative fait main basse sur une grande partie des solutions jugées intuitivement de meilleure qualité pour $\succcurlyeq$, car toute l'information incomparable à un ensemble de choix relativement maximisant est ignorée. Il est généralement nécessaire de considérer un autre axiome, exprimant que l'information n'est partielle que dans des limites données (non redondance, approximation contrôlée, etc.). A la section suivante, un nouveau type d'axiomes exprimant la redondance et l'approximation est présenté.

Plusieurs questions algorithmiques naissent avec ces nouveaux concepts : Existe-t-il un algorithme polynomial en le codage de $(S, \succcurlyeq)$ retournant un e.r.o.m.i. si S est non vide, "non" sinon ? Existe-t-il un algorithme polynomial en le codage de $(S, \succcurlyeq)$ retournant un e.r.o.m.i. de cardinal minimum si S est non vide, "non" sinon ? Les mêmes questions sont valables pour les e.r.m.m.i.. Très importantes pour l'adoption des e.r.o.m.i. et/ou e.r.m.m.i. en tant qu'axiomes

de choix dans la pratique, ces questions ne sont pas traitées ici. Nous avons décidé de continuer à confronter ces concepts aux autres axiomes candidats pour confirmer, infirmer ou nuancer leur intérêt.

3.4.4.5. Optimalité Relative, Maximalité Relative et Domination

Comparons maintenant succinctement l'optimalité relative et la maximalité relative à la domination. Reprenons l'exemple de la Figure 15 : $\{x_1\}$ est l'ensemble maximal, et aussi l'unique e.r.m.m.i.. $\{x_1, x_4\}$ et $\{x_1, x_3\}$ sont les ensembles dominants minimaux. $\{x_1, x_3\}$ est l'unique noyau et $\{x_1, x_4\}$ est l'unique ensemble à la fois relativement maximal, dominant et minimal pour l'inclusion. $\{x_1, x_3, x_4\}$ est l'unique e.r.o.m.i..

Voici quelques propriétés, données de manière informelle, liant ces trois axiomes :
L'optimalité relative implique la domination. C'est-à-dire que tout ensemble relativement optimal est un ensemble dominant. La preuve est assez directe. La réciproque est fausse comme nous avons vu avec l'exemple.
La domination est compatible avec la maximalité relative, c'est-à-dire qu'il existe des ensembles satisfaisant le système axiomatique {domination, maximalité relative}. Dans l'interprétation donnée par le problème des radars[78], ce système axiomatique identifie les sous-ensemble de solutions qui voient tout le monde, sans être vus de l'extérieur. De tels ensembles existent pour tout couple $(S, \succcurlyeq)$; Par exemple les ensembles relativement optimaux.
Notons par **e.r.m.d.m.i.** un ensemble satisfaisant le système axiomatique {domination, maximalité relative} et qui est minimal pour l'inclusion. Un tel ensemble existe pour tout couple $(S, \succcurlyeq)$. Nous ne verrons pas ici les liens existant entre un ensemble optimal ou maximal et les e.r.m.d.m.i.. Remarquons cependant que les e.r.m.d.m.i. sont un intermédiaire intéressant entre les noyaux et les ensembles maximaux. En complément, il existe un e.r.m.d.m.i. non vide pour tout couple $(S, \succcurlyeq)$ tel que S est non vide. Les éléments de S choisis par les e.r.m.d.m.i. vérifient que tout élément non choisi est au plus aussi bon qu'un élément choisi (domination) et qu'aucun élément non choisi ne peut être préféré à une solution choisie (maximalité relative).

3.4.4.6. Conclusion

En définitive, l'optimalité relative et la maximalité relative sont des généralisations 'naturelles' respectivement de l'optimalité et de la maximalité. Ces axiomes s'inscrivent dans une logique d'explication de conflits parmi les solutions choisies. Chaque e.r.o.m.i. (resp. e.r.m.m.i.) identifie un ensemble de solutions de meilleure qualité, qui ne peut être plus restreint à cause des conflits internes[79].
Pour aller plus loin, remarquons qu'en termes de conflit, le noyau, via la stabilité, ne tient compte que des conflits n'impliquant que deux solutions : Si deux solutions sont incomparables alors elles satisfont des propriétés incommensurables et doivent être toutes les deux présentées au décideur. Par contre la stabilité évacue les conflits impliquant plus de deux solutions dans des circuits de préférence stricte, contrairement aux e.r.o.m.i. et e.r.m.m.i (qui mettent ces circuits en valeur).

---

[78] Le problème des radars (Berge [1983, chap. 14]) interprète l'ensemble S comme un ensemble de points stratégiques, l'assertion "$x \succcurlyeq y$" par "un radar placé au point stratégique x peut surveiller le point stratégique y". Le but est alors d'identifier un sous-ensemble de points stratégiques où placer des unités militaires pourvues de radars pour effectuer une action bien précise. Par exemple, la surveillance de l'ensemble S ($\Leftrightarrow$ identifier un sous-ensemble dominant).

[79] De même que des problèmes d'explication d'inconsistance ont été définis dans le cadre de la faisabilité, des problèmes d'explication d'inconsistance préférentielle et d'explication de conflits peuvent être formulées. Ces axiomes s'inscrivent ainsi exactement dans la continuité de l'idée de SEN et FOSTER : Si un concept est ambigu, présenter cette ambiguïté (voir citation en début de chapitre).

Ces travaux axiomatiques illustrent bien l'énorme forêt qui reste à défricher. Ils permettent aussi de ne pas oublier – voire *admettre* – que la caractérisation des solutions choisies est elle-même un problème, lorsque l'on sort des sentiers battus de la transitivité. Pour clôturer ce paragraphe, la Figure 18 récapitule la relation d'inclusion entre les axiomes.

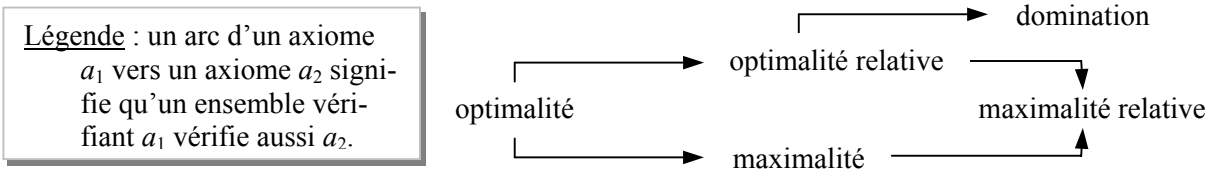


Figure 18 : Synthèse de la relation d'inclusion entre les propriétés de qualité globale.

Présentons maintenant un autre axiome, l'indiscernabilité parfaite, véhiculant plus clairement que la stabilité, la notion de redondance.

3.4.5. Redondance Exprimée par l'Indiscernabilité

La pratique de l'aide à la décision nous a appris qu'elle va de pair avec une information concise, non redondante (§ 3.2.2.5). Elle a été véhiculée presqu'exclusivement par la notion de stabilité. Pourtant, sorti du cadre transitif, cet axiome peut se montrer très intransigeant et arbitraire dans son choix de restriction de l'information. En recherchant uniquement des solutions incomparables, la stabilité supprime à la fois l'information redondante et effectue une approximation incontrôlée, comme nous allons le voir.

L'IA[80] et la philosophie[81] ont apporté de nouveaux arguments pour renouveler les axiomes généraux de choix de solutions satisfaisantes. Ainsi, la redondance a pris plusieurs formes comme l'indiscernabilité, la similitude ou le *ceteris paribus* (toute chose égale par ailleurs). Nous nous limitons ici principalement à la propriété d'indiscernabilité parfaite :
Étant donné un ensemble non vide S structuré par une relation binaire de préférence $\succcurlyeq$, deux solutions x et y de S sont **parfaitement indiscernables**[82] pour $\succcurlyeq$ ssi :

$$\text{Pour tout } z \in S,\ \ x\ \alpha\ z \Leftrightarrow y\ \alpha\ z \text{ avec } \alpha \in AF(RBP) = \{\succ, \prec, \simeq, \|\}\ . \qquad (3\text{–}12)$$

(parfaite indiscernabilité)

Cette propriété définit la relation d'indiscernabilité parfaite PI($\succcurlyeq$) sur S :

$$x\ \mathrm{PI}(\succcurlyeq)\ y \Leftrightarrow p_{\succcurlyeq}(x, z) = p_{\succcurlyeq}(y, z) \text{ pour tout } z \in S\ ,$$

en utilisant la fonction de préférence $p_{\succcurlyeq}$ associée à la relation binaire de préférence $\succcurlyeq$ comme définie au § 3.3.2.1 : $p_{\succcurlyeq}(x, y) = \alpha \Leftrightarrow x\ \alpha\ y$ avec $x, y \in S$ et $\alpha \in AF(RBP) = \{\succ, \prec, \simeq, \|\}$.

La relation qu'entretient x avec les éléments de S est équivalente à celle qu'entretient y avec les éléments de S. En d'autres termes, x et y sont du même avis sur tous les éléments de S. Lorsque deux solutions sont parfaitement indiscernables, il suffit de connaître l'opinion d'une solution pour connaître l'opinion de l'autre. Par conséquent lors d'un choix des

[80] Par exemple, en Modélisation et Représentation des Préférences, voir les travaux de BOUTILIER *et al*. [1997], BOUTILIER *et al*. [1999], LAFAGE & LANG [2001], LANG [2002]. La théorie des ensembles incertains (*rough sets theory*) est basée sur un ensemble d'axiomes véhiculant aussi ce type d'information (*A*-indiscernabilité, similarité, etc.). Nous en reparlerons dans la suite de cette sous-section.

[81] Par exemple, le professeur d'économie et de philosophie Amartya SEN, dans son livre [1970, chap. 9, p. 134], utilisa le concept de similarité pertinente (*relevant similarity*) : « *If x and y are exactly similar except in some respects, and if a person's jugements in question involving x and y are independent of those respects, then x and y are relevantly similar in that person's system* »

[82] En AMCD, ROY & BOUYSSOU [1993 § 6.1.2 c) et § 6.2], ou encore AÏT YOUNES *et al*. [2000*a*], [2000*b*], parlent de **vrais ex æquos**.

meilleures solutions, nous pouvons nous restreindre à une solution dans chaque classe de parfaits indiscernables. L'indiscernabilité parfaite PI($\succcurlyeq$) satisfait un ensemble de propriétés :

Proposition 9 : Étant donné un ensemble non vide structuré par une relation binaire de préférence (S, $\succcurlyeq$), alors
(a) PI($\succcurlyeq$) est une relation d'équivalence sur S
(b) x PI($\succcurlyeq$) y $\Rightarrow$ x et y sont indifférents
(c) Si $\succcurlyeq$ est transitive, alors (indifférence $\Leftrightarrow$ indiscernabilité parfaite).
(d) La transitivité de l'indifférence n'implique pas l'équivalence entre l'indiscernabilité parfaite et l'indifférence. En définitive, Si (indifférence $\Leftrightarrow$ indiscernabilité parfaite), alors $\succcurlyeq$ est transitive

Preuve : Voir en Annexe B1.

En résumé, l'ensemble des couples d'éléments parfaitement indiscernables est inclus dans l'ensemble des couples d'éléments indifférents. Dire que deux solutions sont indifférentes, n'est pas équivalent à dire que ces solutions sont parfaitement indiscernables. Les assertions (c) et (d) informent que la transitivité est la propriété minimale de $\succcurlyeq$ pour qu'il y ait équivalence entre l'indifférence et l'indiscernabilité parfaite.

Maintenant, voyons comment cet axiome peut s'appliquer à la caractérisation d'un ensemble de choix. Étendons cette propriété aux ensembles :

Étant donné le couple (S, $\succcurlyeq$) et deux sous-ensembles non vides A et B de S, alors :

- A est PI($\succcurlyeq$)-contenu dans B, ssi l'information préférentielle concernant A dans (S, $\succcurlyeq$) est contenue dans celle de B dans (S, $\succcurlyeq$) ; c'est-à-dire :

$$\forall\, x \in A, \exists\, z \in B \text{ tel que } x \text{ PI}(\succcurlyeq)\, z\,.$$

- A et B sont **parfaitement indiscernables** ssi :

$$\forall\, x \in A, \exists\, z \in B \text{ tel que } x \text{ PI}(\succcurlyeq)\, z, \qquad \text{et} \qquad \forall\, y \in B, \exists\, z \in A \text{ tel que } y \text{ PI}(\succcurlyeq)\, z\,.$$

En d'autres termes, A et B sont parfaitement indiscernables si et seulement si A est PI($\succcurlyeq$)-contenu dans B et B est PI($\succcurlyeq$)-contenu dans A.

Le sous-ensemble A de S est **minimal pour l'inclusion selon l'indiscernabilité parfaite**, s'il n'existe pas de sous-ensemble strict de A qui lui soit parfaitement indiscernable. Par conséquent, une autre manière de répondre à un problème d'aide au choix à partir d'une relation binaire de préférence, est de définir l'ensemble de choix comme étant un sous-ensemble de l'ensemble optimal $B(S, \succcurlyeq)$ ou de l'ensemble maximal $M(S, \succcurlyeq)$, qui plus est minimal pour l'inclusion selon l'indiscernabilité parfaite. De tels ensembles sont notés respectivement $IPB(S, \succcurlyeq)$ et $IPM(S, \succcurlyeq)$. L'utilisation de l'indiscernabilité parfaite peut aussi s'appliquer à des e.r.o.m.i. ou des unions d'e.r.o.m.i., ou encore à des e.r.m.d.m.i. ou des unions d'e.r.m.d.m.i..

L'indiscernabilité parfaite est une manière d'exprimer la redondance d'information préférentielle. Sortie du cadre de la transitivité, elle peut cependant paraître peu réductrice en quantité d'information. L'ajout d'axiomes exprimant l'approximation est une possibilité. Mais n'oublions pas que l'interaction avec le décideur est aussi une alternative de très grande valeur. La stabilité maximale a été presqu'exclusivement utilisée pour exprimer à la fois la redondance et l'approximation. Voici quelques propriétés liant les ensembles de choix optimisants (resp. maximisants) stables maximaux, et les ensembles optimisants (resp. maximisants) minimaux pour l'inclusion selon l'indiscernabilité parfaite. De manière informelle, nous avons :

Si $\succcurlyeq$ est transitive, alors A est un $SB(S, \succcurlyeq) \Leftrightarrow A$ est un $IPB(S, \succcurlyeq)$. De même, A est un $SM(S, \succcurlyeq) \Leftrightarrow A$ est un $IPM(S, \succcurlyeq)$. Cependant, pour des relations binaires de préférences plus faiblement structurées, ces axiomes sont très souvent incompatibles et représentent deux possibili-

tés divergentes de la notion de meilleure qualité. Chacune pousse à une caractérisation différente. Remarquons quand même que pour tout $IPB(S, \succcurlyeq)$, il existe un $SB(S, \succcurlyeq)$ tel que $SB(S, \succcurlyeq) \subseteq IPB(S, \succcurlyeq)$ et pour tout $IPM(S, \succcurlyeq)$, il existe un $SM(S, \succcurlyeq)$ tel que $SM(S, \succcurlyeq) \subseteq IPM(S, \succcurlyeq)$. Les démonstrations sont immédiates.
Ces propriétés montrent que la stabilité supprime l'information redondante, exprimée par l'indiscernabilité parfaite. Par contre, en dehors d'une structure transitive, elle supprime des solutions optimales ou maximales non redondantes. La suppression de cette information non redondante dépend essentiellement de l'algorithme qui choisi arbitrairement un sous-ensemble de solutions optimales ou maximales incomparables entre elles.

L'approximation peut être modélisée d'autres manières. Par exemple, en théorie des ensembles incertains (*rough sets theory*), elle a pris la forme principalement de l'A-indiscernabilité (voir par exemple GRECO *et al*. [2001]). Appliqué à notre problème de caractérisation, l'A-indiscernabilité se formule comme suit :

Étant donné un couple $(S, \succcurlyeq)$, et un sous-ensemble non vide A de S, deux éléments x et y de S sont ***A*-indiscernables** ssi : $p_{\succcurlyeq}(x, z) = p_{\succcurlyeq}(y, z)$ pour tout $z \in A$.

Les éléments x et y sont du même avis concernant les éléments de $A \subseteq S$. L'indiscernabilité parfaite est donc un cas particulier de ces relations et correspond à la S-indiscernabilité. L'indiscernabilité parfaite exprime la redondance, alors que l'A-indiscernabilité exprime l'approximation. L'approximation devient alors une distance, un écart entre les sous-ensembles A de S et l'ensemble S lui-même. L'indiscernabilité est donc une alternative à la stabilité maximale pour décrire la redondance et l'approximation. L'indiscernabilité a, par rapport aux autres axiomes, l'avantage de séparer la redondance de l'approximation. Remarquons que d'autres choix que l'A-indiscernabilité sont possibles pour modéliser l'approximation (GRECO *et al*. [2001]). Celle-ci peut également ne pas être prise en compte.

Pour conclure, cette sous-section, la Figure 19 synthétise les liens d'inclusion existants entre les axiomes présentés.

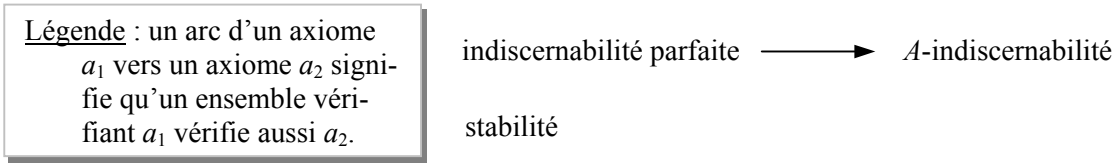


Figure 19 : Synthèse des relations d'inclusion des propriétés de type redondance et approximation.

3.4.6. Conclusion et Compléments

3.4.6.1. Conclusion

Dans cette section, nous nous sommes intéressés aux axiomes d'identification des solutions de meilleure qualité, à partir d'une unique relation binaire de préférence. Le passage de la relation binaire de préférence (préférences données sur des couples de solutions alternatives), vers l'ensemble de choix (un choix global) peut se faire de diverses manières. Cette diversité fait du choix des axiomes de choix d'abord un problème 'multicritère' : Plusieurs axiomes, souvent partiellement compatibles, représentent un même concept comme la meilleure qualité globale, la redondance, l'approximation, etc.. Lors d'inconsistance, un choix des axiomes s'impose au concepteur de procédures d'aide au choix, selon la priorité qu'il accorde aux notions sous-jacentes. Ce choix peut être grandement facilité si le concepteur tient compte du fait que cette problématique est technique, et donc s'insère dans tout un processus décisionnel. En allant plus loin, il est possible de concevoir des systèmes informatiques, permettant à l'utilisateur (ou décideur) de prendre part à ce choix. Malheureusement, de par son niveau

élevé d'abstraction, de tels systèmes pourront se trouver dénués d'intérêt pour le décideur néophyte. Quoi qu'il en soit, il est nécessaire de faire un travail de comparaison des axiomes entre eux, et de les confronter à des exemples d'école pertinents pour tester leur validité et éclaircir les liens entre axiomes. En d'autres termes, une analyse axiomatique est nécessaire.

3.4.6.2. Ensemble de Rejet

Au lieu de s'intéresser à l'ensemble de choix, l'intérêt du décideur peut porter sur l'ensemble de rejet dans S, ou le complémentaire de l'ensemble de rejet dans S. L'**ensemble de rejet** de $(S, \succcurlyeq)$ associé à un système axiomatique de choix, est l'ensemble de choix de $(S, \succcurlyeq')$ associé au même système axiomatique, tel que :

$$\text{Pour } x, y \in S,\ x \succcurlyeq' y \Leftrightarrow y \succcurlyeq x$$

Très connu en TCS, l'ensemble de rejet n'a suscité, à notre connaissance, que très peu de travaux en AMCD (voir BANA E COSTA [1993]). Faisons remarquer que certains axiomes caractérisant l'ensemble de choix et l'ensemble de rejet peuvent comporter des éléments en commun. C'est le cas par exemple du noyau (HAYNES et *al.* [1998*a*] et [1998*b*]). Il en est de même de la maximalité, par exemple lorsqu'il existe des éléments incomparables ou indifférents à tous les autres. On parle alors du **paradoxe du choisi-rejeté**. Dans le contexte des procédures de vote on parle du paradoxe du vainqueur-vaincu (voir DURAND [2000], RAYNAUD [2000]).

3.4.6.3. Axiomes de Choix et Relations Non Binaires de Préférence

Remarquons que ces axiomes permettent de construire des relations non binaires de préférence (§ 3.3.4) sur S. Étant donné $(S, \succcurlyeq)$ et un système axiomatique de choix SA, alors une relation non binaire de préférence peut être définie comme suit :

$$rnp_{SA,\succcurlyeq}(S) = (\mathcal{A}, C_{SA,\succcurlyeq}) \text{ avec } \mathcal{A} = \{A \in \mathcal{P}(S) \setminus \{\varnothing\} \text{ tel que : } C_{SA}(A, \succcurlyeq_A) \neq \varnothing\} \qquad (3\text{–}13)$$
$$\text{et } C_{SA,\succcurlyeq}(A) = C_{SA}(A, \succcurlyeq_A) \text{ pour tout } A \in \mathcal{A} ;$$

$\succcurlyeq_A$ étant la restriction de $\succcurlyeq$ à A et C_{SA} l'ensemble de choix défini par SA.

En d'autres mots, si nous possédons une relation binaire de préférence sur un ensemble non vide S et un système axiomatique de choix, on peut identifier un ensemble de choix sur n'importe quel sous-ensemble non vide de S, si cette définition est acceptée.

La conception de relations binaires de préférences à partir de relations non binaires de préférence a donné lieu à la théorie des préférences révélées. Nous en reparlerons aux § 5.3.4 et 5.3.5.

3.4.6.4. Identification de Problèmes Pratiques d'Aide au Choix

L'identification de l'ensemble de choix est la première étape sur laquelle s'appuie la définition des problèmes qui apporteront une réelle aide à la décision. Voici quelques exemples de problèmes :

- Trouver un ensemble de choix si le système axiomatique identifie un sous-ensemble non vide, sinon retourner "non" ;
- Trouver une solution de l'ensemble de choix si S est non vide, sinon retourner "non" ;
- Proposer des outils de parcours de l'ensemble de choix.
- Proposer des outils permettant de connaître les propriétés de l'ensemble de choix. Par exemple, quel est sa cardinalité ? Quels sont les axiomes structurels vérifiés : Est-il structuré en stable ? De combien de composantes fortement connexes est-il composé ? etc.

L'interactivité devient un passage obligé pour tout problème dont l'instance $(S, \succcurlyeq)$ est donnée en compréhension (problème combinatoire ou continu), et lorsque l'optimalité ne peut être atteinte.

Remarquons aussi que la suppression en premier abord de l'information redondante dans les ensembles de choix (mettant en œuvre la notion de filtre intelligent de l'information (LÉVINE & POMEROL [1989])) ne doit pas faire oublier que l'utilisateur peut vouloir explorer les éléments de *S* jugés similaires à un élément donné. Car même si deux solutions sont identiques du point de vue des préférences modélisées, certaines nuances peuvent avoir échappé à l'instruction du modèle préférentiel. L'utilisateur pourra de cette manière rectifier cette perte en bonne et due forme[83].

Dans les problèmes réels, la relation binaire de préférence est généralement une information préférentielle obtenue en synthétisant les préférences de plusieurs individus ou/et points de vue d'un individu. C'est le sujet de notre prochain chapitre.

[83] Ce protocole rejoint la problématique technique de la robustesse. Nous renvoyons à ROY & BOUYSSOU [1993], DURAND [2000] pour des problèmes liés à cette problématique.

4. Avancées en Modélisation : Profils, Information Interindividuelle et Agrégation

Ce chapitre est consacré aux modèles préférentiels où interviennent plusieurs acteurs usant de plusieurs critères pour formuler leurs préférences. La section 4.1 fait une revue des principaux modèles utilisés dans la littérature. Les notions de profil et de coalition sont présentées au § 4.1.1, suivies des modes usuels de représentation de l'information préférentielle interindividuelle au § 4.1.2. Nous clôturons cette section, par l'introduction des profils étendus basés sur les comparaisons interpersonnelles directes. Ces profils sont de plus en plus utilisés en sciences humaines pour traiter de problèmes éthiques de décision publique (allocation de ressources publiques), ou de justice et politique théoriques. Les comparaisons interpersonnelles sont les fondements nécessaires à la définition de l'équité. La section 4.2 est consacrée aux outils de traitement de cette information préférentielle dans le cas de la problématique d'aide au choix. Le concept de règle d'agrégation est exposé, ainsi que les méthodes d'analyse axiomatique. Ces méthodes fournissent une aide théorique, mais aussi pratique, au choix des règles identifiant les solutions les mieux adaptées au problème étudié.

4.1. Profils et Informations Interindividuelles

Jusqu'à maintenant nous n'avons fait aucune référence à l'émetteur des préférences. Or dans un système (problème où plusieurs solutions potentielles doivent être évaluées), différentes personnes peuvent être amenées à faire part de leur opinion ; chaque personne pouvant évaluer les solutions potentielles selon plusieurs points de vue. C'est dans ce cadre que la notion de profil est apparue.

4.1.1. Le Profil Simple

4.1.1.1. Profil Simple et Individu

Un **profil simple** $\mathcal{F}$ d'un ensemble d'individus I_e sur un ensemble de solutions potentielles S est un ensemble fini en extension de préférences sur S appelées préférences individuelles, associant à chaque individu i de I_e ses préférences $pref(i, S)$ sur S :

$$\mathcal{F}(I_e, S) = \{pref(i, S), \text{ pour tout } i \in I_e\}$$

Dès ici, les profils peuvent se distinguer suite à la définition d'un individu. Le terme d'**individu** est employé pour désigner aussi bien un point de vue particulier émis par une personne ou par un groupe de personnes, que l'opinion générale d'une personne ou d'un groupe de personnes. Ainsi, un profil peut être situé :

(a) au niveau d'une personne. Les individus de cette personne désignent alors ses différents critères de jugement des solutions potentielles, le but pouvant être alors de concevoir les préférences globales de cette personne ;

(b) au niveau d'un groupe. Un individu s'identifie à une personne du groupe. Le but peut être alors la recherche de préférences sociales ;

(c) à un niveau supérieur, pour établir par exemple les préférences globales de plusieurs communautés, entre différentes solutions. Une communauté désigne un individu ; ou
(d) à des niveaux intermédiaires, voire transversaux (du moment que l'étude de tels profils a un sens pour les acteurs concernés).

En AMCD, l'individu désigne traditionnellement le critère, alors qu'en TCS classique, l'individu s'identifie à la personne, la communauté, le pays, etc.
Une définition plus formelle, basée sur la notion de point de vue, peut être donnée : Considérons un ensemble de personnes E, et pour chaque sous-ensemble non vide de E un hypothétique ensemble de points de vue $PDV(E_1)$, pour tout $\emptyset \neq E_1 \subseteq E$; un individu est un élément de $\{(E_1, pdv)$ tel que $E_1 \subseteq E$, $E_1 \neq \emptyset$ et $pdv \in PDV(E_1)\}$.
En pratique, un individu s'identifie très facilement, car il est dépendant du contexte.

Dans ce document, nous notons $\mathcal{P}(\mathsf{PREF}, S)$ l'ensemble des profils simples sur S dont les préférences individuelles sont structurées selon l'axiomatique PREF. Par exemple, $\mathcal{P}(\mathsf{POP}, S)$ est l'ensemble des profils de préordres partiels, si on utilise la notation du § 3.3.2.1.

Nous nous plaçons maintenant, et ce jusqu'à la fin de cette section, dans le cadre de préférences ayant une cohérence de choix local binaire : les préférences binaires (§ 3.3.1 à 3.3.3).

4.1.1.2. Conflits Locaux, Coalitions Concordante et Discordante

Supposons que les préférences individuelles d'un profil simple $\mathcal{F}(I_e, S)$ ont la même structure de préférence :

$$SP(pref(i, S)) = SP \quad \text{pour tout } i \in I_e$$

On note AF, l'ensemble des attitudes fondamentales de SP.

Le profil simple $\mathcal{F}(I_e, S)$ peut être représenté d'une autre manière, en associant à chaque choix local l'ensemble des individus qui ont effectué ce choix dans leurs préférences. Nous parlons alors de coalition.
Ainsi, la **coalition** $CO(x\ \alpha\ y)$ **d'individus concordante avec la proposition** $x\ \alpha\ y$, avec (x, y) un couple de solutions potentielles de S et α une attitude de $\mathcal{PR}(AF)$, est définie par :

$$CO(x\ \alpha\ y) = \{i \in I_e \text{ tel que } p_i(x, y) \subseteq \alpha\}$$

La notion de coalition permet de représenter un profil simple $\mathcal{F}(I_e, S)$ d'une autre manière, comme l'ensemble des coalitions d'individus concordants avec les différents choix locaux possibles :

$$\mathcal{F}(I_e, S) = \{CO(x\ \alpha\ y) \subseteq I_e \text{ pour tout } \alpha \in \mathcal{PR}(AF) \text{ et pour tout } (x, y) \in S \times S\}$$

Dans le cas particulier des relations de préférence valuées (§ 3.3.2.2), la structure de l'ensemble des attitudes peut être exploitée pour intégrer de l'incertitude dans la définition de la notion de coalition concordante (voir PERNY [1992] et PERNY [1998]).

La notion de discordance est un peu plus difficile à définir car, ne pas être en concordance avec une proposition $x\ \alpha\ y$, ne signifie pas pour autant, être en net désaccord avec elle, comme le faisaient remarquer ROY & BOUYSSOU [1993 § 3.1.2]. Par exemple, si la structure de préférence SP renferme des attitudes fondamentales exprimant différentes manières d'apprécier la préférence, ce lien influence la définition de la discordance. Lorsque pour un couple de solutions $x, y \in S$ nous avons : $p_i(x, y) \cap p_j(x, y) = \emptyset$ pour deux individus i, j disjoints, alors nous dirons qu'il existe un **conflit local** entre i et j. Le conflit est plus ou moins important selon l'information véhiculé par chaque attitude et les relations entre attitudes. Le conflit peut être bénin, faible, présumé favorable à l'un des concernés, ou fort. Dans le cadre de la relation

binaire de préférence, la **coalition** $CD(x \succcurlyeq y)$ **d'individus discordante avec la proposition** $x \succcurlyeq y$, avec (x, y) un couple de solutions potentielles de S est définie par :

$$CD(x \succcurlyeq y) = \{i \in I_e \text{ tel que } p_i(x, y) = \prec\}.$$

Pour des définitions plus générales, nous renvoyons à ROY & BOUYSSOU [1993] et PERNY [1992], [1998].

4.1.1.3. Attitudes Fondamentales d'un Profil Simple

Considérons un profil simple $\mathcal{F}(I_e, S)$ ayant une cohérence de choix local binaire. Notons $AF(i)$, l'ensemble des attitudes fondamentales associé aux préférences de l'individu i sur les solutions de S. L'**ensemble des attitudes fondamentales** $AF(\mathcal{F})$ **du profil simple** $\mathcal{F}$ est le produit cartésien des ensembles d'attitudes fondamentales individuelles de $\mathcal{F}$:

$$AF(\mathcal{F}) = \bigtimes_{i \in I_e} AF(i)$$

En terme de dénombrement, le nombre de sommets de l'ensemble $AF(\mathcal{F})$ est égal au produit des cardinaux des ensembles $AF(i)$. Dans le cas où le nombre d'attitudes fondamentales individuelles est fini et égal pour tous à une constante m, alors $\mathrm{card}(AF(\mathcal{F})) = m^{\mathrm{card}(I_e)}$.

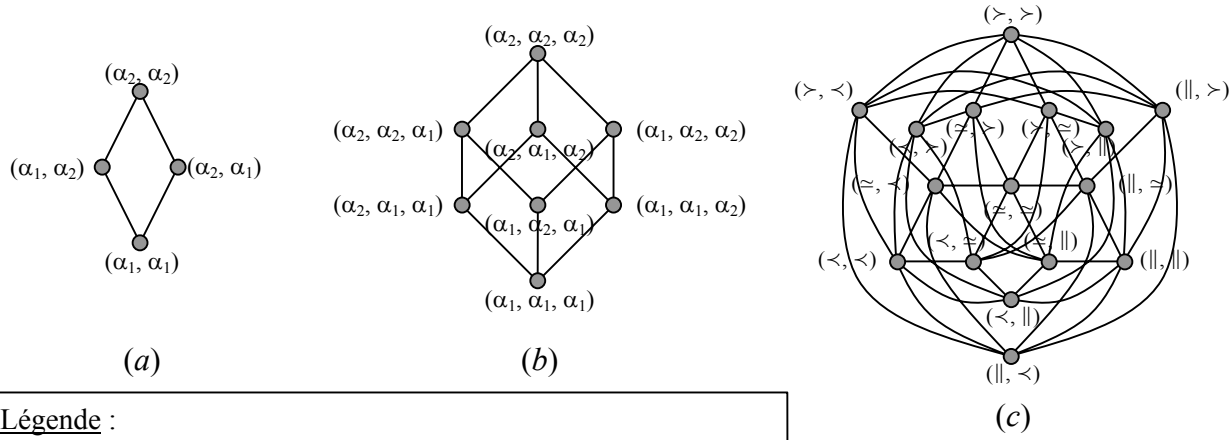
L'ensemble des attitudes d'un profil simple $\mathcal{PR}(AF(\mathcal{F}))$ est l'ensemble des parties de $AF(\mathcal{F})$ privé de l'ensemble vide et de $AF(\mathcal{F})$ $\Leftrightarrow$ $\mathcal{PR}(AF(\mathcal{F})) = \mathcal{P}(AF(\mathcal{F})) \setminus \{\varnothing, AF(\mathcal{F})\}$.

L'ensemble des attitudes fondamentales d'un profil peut servir de base à l'étude de relations binaires entre ces éléments. Son utilisation peut s'avérer intéressante notamment dans les études de manipulabilité et dans la conception des préférences collectives de qualité.

Exemple d'utilisation : Définissons sur l'ensemble des attitudes fondamentales d'un profil simple $\mathcal{F}$, la relation binaire CA_1 de changement d'attitude d'un seul individu :

$$\text{Pour tout } \alpha_1, \alpha_2 \in AF(\mathcal{F}), \quad \alpha_1 \; CA_1 \; \alpha_2 \Leftrightarrow \begin{cases} \exists i \in I_e \text{ tel que } \alpha_1(i) \neq \alpha_2(i) \text{ et que} \\ \alpha_1(j) = \alpha_2(j) \text{ pour tout } j \in I_e \text{ et } j \neq i \end{cases}$$

Cette relation exprime les transitions possibles entre les attitudes, lorsqu'un seul individu change d'avis. CA_1 est symétrique. Par conséquent, on peut le représenter par un graphe non orienté simple. La Figure 20 montre 3 exemples de graphe.



Légende :
(*a*) représente le graphe associé à la relation CA_1 pour $I_e = \{i, j\}$, $AF(i) = AF(j) = \{\alpha_1, \alpha_2\}$.
(*b*) représente le graphe associé à la relation CA_1 pour $I_e = \{i, j, k\}$, $AF(i) = AF(j) = AF(k) = \{\alpha_1, \alpha_2\}$.
(*c*) représente le graphe associé à la relation CA_1 pour $I_e = \{i, j\}$, $AF(i) = AF(j) = \{\succ, \prec, \simeq\}$.

Figure 20 : Illustrations de la relation CA_1 de changement d'attitude d'un seul individu.

Autres propriétés de CA_1 : Dans le cas particulier où les individus ont même ensemble d'attitudes fondamentales ($\Leftrightarrow AF(i) = AF$, pour tout $i \in I_e$), celui-ci étant fini de cardinal m, alors nous obtenons un graphe non orienté simple, régulier de degré $\mathrm{card}(I_e)\cdot(m - 1)$, d'ordre $m^{\mathrm{card}(I_e)}$, et de diamètre $\mathrm{card}(I_e)$.
Certains de ces graphes ont fait l'objet de beaucoup de travaux (les hypercubes (AICHHOLZER [1997]) par exemple, dont les graphes de la Figure 20(*a*) et (*b*) en sont des représentants). Des études ont aussi été menées sur la définition récursive des graphes[84], facilitant ainsi la généralisation de résultats, obtenus sur de petits exemples, à un nombre quelconque d'individus ou d'attitudes.
Bien entendu la relation CA_1 peut se généraliser au changement d'attitude de plusieurs individus.

Ces graphes sont une manière d'étudier la manipulabilité de certaines décisions collectives, que peut exercer un (ou un groupe d') individu(s). Ils peuvent aussi servir à créer de nouvelles règles collectives. Beaucoup de choses sont encore à dire sur le sujet, mais nous en resterons là dans ce document.

Pouvons-nous aller plus loin, et parler de structure de préférence d'un profil simple ? En général oui : Illustrons nos propos. L'ensemble des attitudes fondamentales $AF(\mathcal{F})$ du profil simple $\mathcal{F}$ est le produit cartésien des $AF(i)$, $\forall\ i \in I_e$. Cela suggère de prendre comme structure de préférence du profil, la structure minimale produit[85]. Cette suggestion informe déjà qu'une certaine forme d'exhaustivité est acceptée (souveraineté des acteurs d'un problème, etc.). ROY & BOUYSSOU [1993, chap. 2] ont énoncé un ensemble de principes (exhaustivité, cohérence et stabilité) qui paraissent naturels pour identifier un profil de fonction-critères acceptables pour fonder les évaluations globales.

4.1.2. Informations Préférentielles Interindividuelles

4.1.2.1. Définition

Des informations préférentielles complémentaires sur l'environnement d'un profil peuvent être jugées potentiellement intéressantes pour asseoir les décisions collectives. Le fait de dire par exemple que : « si les individus sont du même avis sur le choix d'une solution, alors cette solution est adoptée », est déjà une information liant les préférences individuelles[86]. Dans cette section nous discutons des structures utilisées pour les représenter.
Pour le but fixé, il est quelques fois jugé nécessaire de faire intervenir dans l'évaluation :

- Des comparaisons entre les individus en terme d'importance relative ou de rôle (pouvoir d'influence sur les décisions) de chaque individu par rapport à l'idée directrice du profil (§ 4.1.2.2).
- Des comparaisons d'attitudes entre individus (ROY & BOUYSSOU [1993, § 3.1.4 c) à 3.1.4. e)].).
- Des seuils de veto (ROY & BOUYSSOU [1993, § 3.1.3]) ou des taux de substitution (VINCKE [1989, § III.14], ROY & BOUYSSOU [1993, § 3.1.5]).
- Des comparaisons interpersonnelles de type sympathie étendue (§ 4.1.3).
- Etc..

[84] Par exemple les produits de graphes, et plus particulièrement le produit cartésien de graphes (CHARTRAND & LESNIAK [1996], KOBEISSI [2001]), s'appliquent très bien à la relation CA_1.
[85] La structure minimale produit (ou système de relations produit : REINHARDT & SOEDER [1997, p. 37]) indique que la structure collective possède les propriétés communes à chaque structure individuelle.
[86] Elle satisfait le principe d'exhaustivité qui a vient d'être introduit. Pour une étude conséquente sur ces problèmes, voir ROY & BOUYSSOU [1993, § 2.2.2].

Ces informations ont pour point commun d'utiliser les individus du profil simple non plus – seulement – comme émetteurs du message, mais sujets d'évaluations relatives de manière directe ou non. Ces comparaisons et axiomes d'utilisation partielle sont alors appelées **informations préférentielles interindividuelles**[87].

L'interprétation de ces informations est primordiale, car elle donne des indications sur le mode d'emploi de toutes ces comparaisons. Comment s'imbriquent-elles ? Comment peuvent-elles, au moins partiellement, être utilisées pour des décisions globales ? De plus, elles sont souvent déjà une synthèse d'autres informations préférentielles[88].

Le profil simple, enrichi de telles informations, permet :

- d'aboutir à des profils plus évolués (comme nous le détaillerons dans le reste de la section) ;
- de structurer l'ensemble des attitudes fondamentales du profil pour former une structure de préférence de profil. Ses informations créent souvent des relations d'équivalence entre attitudes, simplifiant l'ensemble des attitudes fondamentales du profil.

❖ Considérations Éthiques

Plusieurs questions peuvent être source de préférences : « Qu'est-ce qui est meilleur pour moi ? », « Qu'est-ce qui est meilleur pour la société ? », « Qu'est-ce qui est meilleur pour tel sous-groupe auquel j'appartiens ? », « Ma situation est-elle aussi bonne que celle des autres ? ». La première question a été traitée par la notion de profil simple. Maintenant les autres questions sont aussi importantes et peuvent compléter de façon pertinente le modèle préférentiel. Elles peuvent donner de réelles informations exploitables pour asseoir des décisions collectives. Les réponses à ces questions forment les informations préférentielles interindividuelles.

Certaines de ces questions soulèvent des problèmes éthiques, notamment sur la confiance à apporter aux jugements émis par un individu sur un autre. Parmi les acteurs du système à modéliser qui peut émettre de tels jugements ? De telles questions remettent aussi en question notre vision classique de l'homme (l'*homo-oeconomicus*). Le profil simple véhicule l'idée que les jugements d'une personne ne peuvent aller que dans le sens de ces intérêts propres : on ne lui demande d'ailleurs que ce qui est meilleur pour lui. Mais des jugements collectifs peuvent être formulés par les acteurs du profil : Comment chaque acteur du profil voit-il la société en tant qu'entité prenant des décisions ?

Des observateurs extérieurs peuvent aussi intervenir dans la formulation de ces décisions globales. C'est ce qui est d'ailleurs fait dans les plannings, avec les acteurs secondaires comme les syndicats et associations (§ 1.1.2). Nous reviendrons sur d'autres exemples au chapitre 6. Pour des réflexions générales sur l'éthique en TCS voir SEN [1977*b*], [1987].

Présentons maintenant les informations préférentielles liées à l'importance relative des individus. Nous réservons le § 4.1.3 aux informations de type sympathie étendues.

4.1.2.2. Poids et Importance Relative des Individus

C'est le type d'information préférentielle de l'environnement d'un profil qui a été le plus étudié dans la littérature. Dans l'esprit d'un décideur, généralement, lorsque le problème s'y

---

[87] L'information préférentielle interpersonnelle est connue dans la littérature sous le terme d'information intercritère en AMCD (ROY & BOUYSSOU [1993]), et d'information interpersonnelle en TCS (SEN [1970], D'ASPREMONT & GEVERS [2002]).

[88] & BOUYSSOU [1999] ont eux aussi fait la remarque : « *In most of cases, when a decision-maker decides that he is going to use an aggregation procedure (to be chosen), he doesn't know yet if the method that he will use will require weight or binary relation or…* ». Puis ils ajoutent : « *If weights (or other strong structures) are to be used, they should emerge during the use of the agregation procedure, like the indifference thresholds, utility curves, and other parameters.* ».

prête, certains critères sont jugés plus importants que d'autres. Dans un contexte de groupe, l'avis de certaines personnes par leur fonction dans le système, leur expérience, leur rôle, etc. doit peser plus que celui d'autres personnes. Dans le cadre d'un projet international, on peut souhaiter que les pays aient un avis, sur la décision globale finale, proportionnel à l'importance de leur investissement.

❖ Modélisations

La représentation la plus utilisée est la **pondération**, associant à chaque individu $i \in I_e$, un poids $w(i) \in {}^{+*}$. Plus l'individu est jugé important, plus le poids est élevé. Un profil simple muni d'un jeu de poids sur les individus est appelé **profil simple pondéré**.

L'utilisation d'une relation plus faible que celle des poids a été entreprise en AMCD dans la conception de différents logiciels. MELCHIOR (Leclercq [1984], VINCKE [1989, § V.7]) par exemple, utilise un préordre partiel sur I_e, nommé **relation d'importance sur les individus**. Un profil simple muni d'une relation d'importance sur les individus est appelé **profil simple avec relation d'importance**.

Généralement, l'information est plus complexe lorsqu'elle est brute. Ainsi, nous pouvons avoir plusieurs jeux de poids émis par différents individus, et pas nécessairement ceux de I_e. Les raisons sont diverses (acteurs multiples divergents sur les poids à accorder aux individus de I_e, ou encore analyse de robustesse (voir ROY & BOUYSSOU [1993])). Il est souvent important de connaître les émetteurs de cette information. Nous les noterons I_w les individus émetteurs des poids. En définitive, nous obtenons un profil simple émis par un ensemble d'individus I_w, sur un autre ensemble d'individus I_e. Le modèle de préférence est alors un couple de profils, formé d'un profil simple sur les solutions $\mathcal{F}(I_e, S)$ et d'un autre sur les individus $\mathcal{F}(I_w, I_e)$, appelé **profil simple avec profil d'importance**.

En reprenant l'exemple du projet international du début de paragraphe, un pays peut ne plus influencer les décisions collectives uniquement en fonction de son investissement, mais aussi sur des considérations autres comme la surface agricole plantée, etc.

Autre exemple : L'importance d'un décideur par rapport à un autre sur un projet, peut résulter de plusieurs critères.

Quelques fois, au lieu de définir des préférences sur I_e, les préférences sont définies sur $\mathcal{P}(I_e)$. Voir par exemple OTHMANI [1998].

❖ Interprétations

L'interprétation d'une pondération, d'une relation d'importance ou d'un profil d'importance peut varier selon le contexte. Il est nécessaire de connaître comment le profil simple est connecté à ce type d'information.

Les questions d'interprétation soulevées sont très complexes. Elles ont donné lieu à plusieurs travaux, dont la thèse de MOUSSEAU [1993]. Ce type d'information préférentielle interpersonnelle est souvent orienté vers une contribution à la « résolution » d'avis divergents, conflictuels entre les individus d'un profil simple. L'orientation est rarement purement descriptive.

4.1.3. Profils Basés sur les Comparaisons Interpersonnelles Étendues

Pour mieux appréhender les notions éthiques d'égalité, d'équité et d'inégalité humaine, les sciences humaines utilisent de plus en plus des modèles s'appuyant sur les comparaisons interpersonnelles.

4.1.3.1. Positions, Profils et Préférences de Type Sympathie Étendue

Étant donnés un ensemble de solutions alternatives S et un ensemble d'individus I_e, ces modèles considèrent (SEN [1973, p. 14]) des préférences individuelles sur $S \times I_e$, le couple (x, i) s'interprétant comme : la **position** (*station*) de l'individu i (avec ses goûts et ses qualités mentales) dans l'état social x. Ainsi, par exemple, quelque soit $j, k \in I_e$ et $x, y \in S$, l'assertion $(x, j) \succcurlyeq_i (y, k)$ signifie que : Selon les jugements de l'individu i, être mis dans la position de l'individu j dans x est au moins aussi bon que d'être mis dans la position de l'individu k dans y. Les individus de I_e sont qualifiés d'**observés**, ou **concernés** par les jugements[89]. L'individu i est appelé individu **émetteur**. Bien entendu, l'émetteur i d'un message peut ne pas être un individu de I_e.

Par conséquent, le nouveau profil obtenu dépend de trois paramètres (I_{se}, I_e, S) et est appelé **profil type sympathie étendue** des individus I_{se} jugeant les différentes positions des individus de I_e sur l'ensemble des solutions potentielles S :

$$\mathcal{F}(I_{se}, I_e, S) = \{pref(i, S \times I_e), \text{ pour tout } i \in I_{se}\} = \mathcal{F}(I_{se}, S \times I_e)$$

$pref(i, S \times I_e)$ est appelé préférences étendues de l'individu i. Le profil simple usuel $\mathcal{F}(I_e, S)$ est contenu dans $\mathcal{F}(I_{se}, I_e, S)$, lorsque $I_e \subseteq I_{se}$ par la relation fondamentale suivante :

$$pref(i, S) \Leftrightarrow pref(i, S \times \{i\}) \text{ pour les individus } i \in I_e \cap I_{se} . \qquad \textbf{(4-14)}$$

Dans le cas des relations binaires de préférences, les préférences $\succcurlyeq'_i$ sur S sont reliées aux préférences $\succcurlyeq_i$ sur $S \times I_e$ par :

$$x \succcurlyeq'_i y \Leftrightarrow (x, i) \succcurlyeq_i (y, i), \quad \text{avec } x, y \in S . \qquad \textbf{(4-15)}$$

Les profils type sympathie étendue utilisent des informations préférentielles supplémentaires, par rapport aux profils simples, qui sont les comparaisons interpersonnelles[90].

Notons $\mathcal{R}(\mathsf{POT}, S \times I_e)$ et $\mathcal{U}(S \times I_e, \quad)$ respectivement l'ensemble des préordres totaux sur $S \times I_e$ et l'ensemble des fonctions objectives sur $S \times I_e$. Quatre cas particuliers de profils type sympathie étendue ont été étudiés en TCS, en économie politique et en philosophie morale, depuis les travaux de SEN[91] [1970, chap. 7 à 8*]. Nous les présentons ci-dessous :

- Le premier cas considère un profil formé des préférences d'un unique individu émetteur structurées en préordre total[92] :
$$I_{se} = \{i\} \text{ et } pref(i, S \times I_e) \in \mathcal{R}(\mathsf{POT}, S \times I_e)$$
- Le second cas considère un profil où les émetteurs et les concernés sont les mêmes individus, les préférences individuelles étant structurées en préordre total[93] :
$$I_{se} = I_e \text{ et } pref(i, S \times I_e) \in \mathcal{R}(\mathsf{POT}, S \times I_e) \text{ pour tout } i \in I_{se}$$
- Le troisième cas considère un profil où les émetteurs et les observés sont les mêmes individus, les préférences individuelles étant structurées en vrai-critère[94] :

---

[89] En AMCD, deux types d'individus ont été considérés : Ceux qui, de par leur intervention, conditionnent directement la décision en fonction du système de valeurs dont ils sont porteurs (les **intervenants**). Et, ceux qui subissent les conséquences de la décision tenant compte aussi de leurs préférences (les **agis**). Voir par exemple ROY & BOUYSSOU [1993, § 1.1]. Les intervenants et les agis sont analogues respectivement aux émetteurs et aux observés présentés ici.

[90] Un petit dénombrement nous fait remarquer qu'une relation binaire sur S nécessite $n(n-1)/2$ comparaisons, alors qu'une relation binaire sur $S \times I_e$ nécessite $nm(nm-1)/2$ comparaisons, en notant $n = \mathsf{card}(S)$ et $m = \mathsf{card}(I_e)$. Ce qui implique une quantité d'information environs m^2 fois supérieur dans le cas étendu par rapport au cas simple.

[91] Notons cependant les contributions de BASU [1987] et OK [1995] sur les relations de préférence valuées étendues. Elles sont présentées au § 6.2.5.

[92] Voir par exemple SEN [1970, chap. 7 à 8*], ROBERTS [1980*a*], SUZUMURA [1997].

[93] Voir à ce propos SEN [1970, chap. 7 à 8*], ROBERTS [1980*a*], SUZUMURA [1983, ch. 5], [1997].

[94] Se référer par exemple à SUZUMURA [1983, ch. 6].

$$I_{se} = I_e, \quad pref(i, S \times I_e) = (g_i = (u_i, \geq), S \times I_e) \quad \text{et} \quad u_i \in \mathcal{U}(S \times I_e, \quad)$$

avec $\mathcal{U}(S \times I_e, \quad)$ l'ensemble des applications de $S \times I_e$ dans associant à chaque couple (x, j) sa valeur pour l'individu i.

- Le quatrième cas considère un profil formé des préférences d'un unique individu structurées en vrai-critère[95].

$$I_{se} = \{i\} \quad \text{et} \quad pref(i, S \times I_e) = (g_i = (u_i, \geq), S \times I_e) \quad \text{et} \quad u_i \in \mathcal{U}(S \times I_e, \quad)$$

Le profil est appelé **profil étendu** (resp. **profil généralisé**) lorsqu'un seul individu donne son avis (resp. lorsque $I_{se} = I_e$).

Un cadre plus vaste a été adopté dans ce document : celui des préférences individuelles étendues $pref(i, S \times I_e)$ respectant uniquement le principe de cohérence des choix initiaux binaires. Lorsque des divergences avec le cadre commun sont identifiées nous le mentionnons.

4.1.3.2. Interprétations des Comparaisons Interpersonnelles

L'utilisation de profils basés sur les comparaisons interpersonnelles de ce type, peut soulever de vives critiques comme cela s'est d'ailleurs produit en économie du bien-être (voir SEN [1987] pour une très riche discussion), en déniant le caractère « scientifique » de ces comparaisons.

En effet, pour qu'une décision collective puisse être basée sur de telles considérations, il faudrait pouvoir les obtenir empiriquement. Lorsqu'on dit « Je préfère être la personne j plutôt que la personne k dans cette situation (un état social, une solution) », il faudrait avoir l'opportunité (or peut-être le malheur, selon le cas) de devenir j ou k, pour émettre ensuite un jugement objectif (SEN [1973, p. 14]).

Et pourtant, nous effectuons de telles comparaisons fréquemment. Par exemple, dans le domaine de la planification d'horaires de travail, nous n'hésitons pas à comparer les différentes quantités de ressources qu'affecte une solution aux différents acteurs. Ces ressources sont par exemple, le nombre de jours de repos, le nombre de nuits, etc (cf. JAUMARD & al. [2000]). Et je crois que ces comparaisons forment l'essentiel des préférences formant nos critères de planning. Ce que nous faisons ni plus ni moins, c'est de gérer le temps imparti à l'organisation et le temps de non-travail des employés. Les principales applications de ces profils se trouvent 'dans l'exercice d'émission de jugements de bien-être social, par un planificateur, un observateur politique ou n'importe qui'[96].

Contrairement aux profils simples, l'influence exercé par les préférences des émetteurs sur les observés sont directement visibles. Les profils usuels (profils simples) occultent complètement ce type d'information, ou le camouflent sous des poids et des seuils. Il est vrai que la masse d'information à récolter peut être une raison d'opter pour les profils usuels. L'AMCD a transformé le modèle dédié au vote et comité (profil simple) pour définir un modèle adapté à l'aide en milieu organisationnel, en élargissant les ordres individuels aux préférences moins structurées, en admettant des informations sur l'importance interindividuelle plus élaborées, et par conséquent, des méthodes de traitement plus générales. Mais est-ce suffisant ? Il est souvent très difficile de définir les poids. Une comparaison plus approfondie, formelle de ces deux profils serait nécessaire (l'interprétation de l'information devient ici primordiale, et

[95] Voir par exemple SEN [1970, chap. 9 à 9*], [1972], [1977*a*], [1986], HAMMOND [1979], [1991], [2002], ROBERTS [1980*b*], [1980*c*], [1997], D'ASPREMONT Claude & GEVERS [2002].

[96] Tiré de SUZUMURA [1997, p.209], lequel cite Amartya SEN [1986] – Fundation of Social Choice Theory: An Epilogue – *in* ELSTER J. & HYLLAND A. (eds.) – *Fundations of Social Choice Theory: Studies in Rationality et Social Change* – Cambridge university press, Cambridge : 213-248. Dans son article de 1986 référencé dans la bibliographie, SEN [1986, § 7.1] donne quelques références où l'utilisation des profils simples et profils type sympathie étendue sont réalisés.

pourrait nécessiter une comparaison au cas par cas). Nous ne l'effectuons pas dans ce document. Par contre, la modélisation de l'information préférentielle de notre problème d'élaboration de plannings (§ 6.4) utilisera un profil mixte. Nous verrons notamment que son utilisation évite des agrégations souvent admises comme naturelles. En d'autres termes, elle s'attaque à l'idée que l'on se fait en AMCD des critères du planificateur, comme étant des informations données, alors qu'elles constituent une première étape d'agrégation supposée primordiale (comme le faisait remarquer d'ailleurs ROY & BOUYSSOU [1993], avec notamment les notions précurseurs de conséquences et de nuage de conséquences). Pour les profils type sympathie étendue, cette première étape n'est pas nécessaire. Elle est même non conseillée pour nos problèmes. Les critères sur lesquels se basent les profils simples usuels en élaboration de planning, sont des agrégations avancées de l'information préférentielle initiale. Elle constitue alors une approche radicalement différente des approches AMCD usuelles.

Maintenant, en quoi l'explicitation de ces informations peut-elle améliorer les décisions collectives ? Les raisons sont multiples :

- Tout d'abord, les dépendances entre préférences individuelles sont mieux identifiées, pour ensuite en user de manière plus adéquate ;
- La notion de rôle des différents acteurs est mieux intégrée dans le modèle ;
- Les conflits sont mieux identifiés, et par conséquent une meilleure gestion peut s'en suivre ;
- Elles permettent aussi de mieux interpréter les comparaisons interpersonnelles directes (§ 4.1.2.2), qui sont généralement utilisées en AMCD avec des notions comme le veto, pour modéliser les informations interpersonnelles de type sympathie étendue.

En définitive, ces informations pourront mieux orienter les décisions collectives, lorsque le problème s'y prêtera.

L'étude de ces profils n'ont pas donné lieu à des travaux appliqués dans un cadre implicite. Pourtant leur application devrait ouvrir un grand nombre de possibilités, notamment pour les problèmes d'allocation de ressources.

4.1.3.3. Liberté et Axiomes d'Identité et d'Identité Complète

La restriction des préférences d'un individu i de I_{se} sur les couples (x, j) pour toute solution x, est notée $pref(i, S \times \{j\})$. Les préférences individuelles d'un individu i expriment ses comparaisons de bien-être interpersonnelles 'subjectives'. Il n'y a donc aucune raison que la situation suivante s'impose : $pref(i, S \times \{j\}) = pref(j, S \times \{j\})$ pour tout $i \in I_{se}, j \in I_e$.
En d'autres termes, les préférences de j entre les solutions pour lui-même, ne sont pas forcément compatibles avec les préférences préconisées par i pour j. Or, de tels jugements nécessitent que l'on se place dans la position de l'autre. Ce qui implique « non seulement que nous ayons les circonstances objectives de ce dernier, mais aussi que nous nous identifions à l'autre en termes de caractéristiques subjectives » (SEN [1970, p.150]). Si nous acceptons ce principe *a priori* raisonnable, nous sommes obligés de nous restreindre aux préférences individuelles satisfaisant l'axiome d'identité suivante[97] :

$$pref(i, S \times \{j\}) = pref(j, S \times \{j\}) \text{ pour tout } i \in I_{se}, j \in I_e . \qquad \textbf{(axiome d'identité)}$$

Maintenant, la plupart des cas ne satisfont pas ce principe. La littérature en économie du bien-être en présente de très intéressants. Par exemple (tiré de SUZUMURA [1996]), si x et y sont deux états sociaux potentiels, qui ne diffèrent que par le fait que j prend de la drogue dans x alors qu'il n'en prend pas dans y ; alors il est parfaitement acceptable que i et j ne soient pas d'accord sur le jugement à porter, sachant que j est un toxicomane, et juge que (x, j)

[97] Cet axiome est défini dans la littérature dans le cadre d'un profil généralisé. Nous présentons ici une généralisation aux profils type sympathie étendue.

est meilleur que (y, j), et que i ne l'est pas, préférant (y, j) à (x, j). Un profil satisfaisant l'axiome d'identité est aussi traité de profil satisfaisant la contrainte de non-paternalisme[98]. Il est assumé que les gens sont pleinement responsables des choix qui les concernent, ils sont, par conséquent, libres d'émettre les choix personnels à leur convenance. Cet axiome véhicule un principe de liberté très attrayant. Nous verrons au § 6.3.3.2 que cet axiome peut entrer en conflit avec des principes comme l'unanimité.

Un profil formulant des comparaisons interpersonnelles 'objectives', est identifié à un profil satisfaisant l'axiome d'identité ci-dessus et telle que l'axiome suivant soit satisfait[97] :

$pref(i, S \times I_e) = pref(j, S \times I_e)$ pour tout $i, j \in I_{se}, j \in I_e \subseteq I_{se}$. **(axiome d'identité complète)**

Un profil type sympathie étendue, vérifiant l'axiome d'identité complète – et par conséquent $I_e \subseteq I_{se}$ – permet de se ramener à un profil étendu, avec I_{se} correspondant à l'unique émetteur des préférences individuelles.

D'autres études sur l'identification des jugements interpersonnels renfermant une part de subjectivité ont été faites. Voir SUZUMURA [1997] pour une synthèse.

4.1.3.4. Axiomes de Comparabilité

Dans le cadre des fonction-critères (§ 3.3.3.6), nous avons vu que les décisions peuvent être classées selon qu'elles soient influencées ou non par les informations intra-critères (axiomes de mesurabilité). Les décisions collectives peuvent être aussi jugées selon qu'elles usent plus ou moins d'informations interpersonnelles. Ces informations sont appelées **axiomes de comparabilité**. Ils font partie, au même titre que les axiomes de mesurabilité, des **axiomes d'invariance**. Ces axiomes ont été étudiés uniquement sur les profils étendus de vrai-critères, et divers axiomes ont été formulés comme :

- l'axiome de non-comparabilité : la décision est indépendante des valeurs attribuées à deux individus différents ;
- l'axiome de niveau comparabilité : la décision ne dépend que de l'ordre sous-jacent aux valeurs attribuées à deux individus observés ;
- l'axiome de pleine comparabilité : la décision dépend des valeurs relatives entre deux individus observés.

Nous renvoyons aux références précitées pour une formalisation et pour la définition d'autres niveaux de comparabilité.

Remarquons que les différentes informations interpersonnelles ne sont pas exclusives. Ainsi, nous pouvons avoir des profils type sympathie étendue pondérés : un poids (ou plus généralement une relation d'importance) est défini sur l'ensemble I_{se}, exprimant le rôle de chaque individu émetteur dans le système. Nous présentons un tel modèle préférentiel au § 6.3.5.2.
Lors de la modélisation, les comparaisons interpersonnelles peuvent être partielles, en admettant l'incomparabilité, des incertitudes, du flou.

Par la suite, nous noterons $\mathcal{PSE}(\mathsf{PREF}, S, I_e)$, $\mathcal{PE}(\mathsf{PREF}, S, I_e)$ et $\mathcal{PG}(\mathsf{PREF}, S, I_e)$ l'ensemble respectivement des profils type sympathie étendue, étendus et généralisés d'individus sur S et I_e. Les mêmes conventions pour les profils et relations binaires de préférences sont valables ici pour PREF. Ainsi, un profil généralisé de préordres partiels sur S et I_e sera noté $\mathcal{PG}(\mathsf{POP}, S \times I_e)$. Remarquons qu'un profil étendu sur S et I_e étant équivalent aux préférences d'un individu sur le produit cartésien $S \times I_e$, $\mathcal{PE}(\mathsf{PREF}, S, I_e)$ est aussi noté $\mathcal{R}(\mathsf{PREF}, S \times I_e)$. Si besoin est de spécifier les individus émetteurs I_{se} (ou leur nombre), nous ajoutons en second argument I_{se}. Par exemple : $\mathcal{PSE}(\mathsf{PREF}, I_{se}, S, I_e)$ est l'ensemble des profils type sympathie étendue de $\mathsf{card}(I_{se})$ individus émetteurs sur S et I_e.

[98] Cet axiome traduit bien aussi la maxime : « Ma liberté finit là où commence celle de l'autre ».

4.2. Aider à Choisir avec un Profil : Solutions de Meilleur Compromis

Dans cette section, nous étudions comment une aide au choix de solutions de meilleure qualité peut se réaliser, à partir d'un modèle de préférence défini par un profil. Rappelons qu'un profil regroupe les préférences de différents individus souvent en désaccords. Comment tirer partie au mieux du profil et des informations interpersonnelles disponibles pour identifier les solutions de meilleure qualité ? Comment choisir le plus petit sous-ensemble, en n'excluant aucune solution de meilleure qualité ? Mais l'aspect conflictuel inhérent aux différents points de vue, permet-t-il toujours de parler de solution de meilleure qualité ? Généralement non. C'est pourquoi nous adopterons dans le cadre des profils le terme de '**solution de meilleur compromis**' pour désigner les solutions que nous voulons caractériser.

Dans une première sous-section (§ 4.2.1), les moyens utilisés pour rendre plus opérationnelle l'information véhiculée par les profils sont présentés en introduisant le concept d'agrégation d'information préférentielle. Nous nous attardons sur un ensemble d'exemples actuellement très utilisés en TCS et en AMCD. Ensuite (§ 4.2.1.5), nous présentons une méthode, l'analyse axiomatique des règles de décision collective, proposant des bases scientifiques pour sélectionner des solutions de meilleur compromis à partir d'un profil.

4.2.1. Choix Collectif et Agrégation

4.2.1.1. Ensemble de Choix Collectif, Conflit Global

Nous considérons dans cette section 4.2 que l'information initiale est fournie sous deux formes différentes :

- Soit un triplet (S, I_e, $\mathcal{F}(I_e, S)$) formé d'un ensemble non vide de solutions S, d'un ensemble d'individus émetteurs I_e, et d'un profil simple $\mathcal{F}(I_e, S)$ des préférences individuelles de I_e sur S ;
- Soit un quadruplet (S, I_e, I_{se}, $\mathcal{F}(I_{se}, I_e, S)$), formé d'un ensemble non vide de solutions S, d'un ensemble d'individus émetteurs I_{se}, d'un ensemble d'individus observés I_e et d'un profil étendu $\mathcal{F}(I_{se}, I_e, S)$ des préférences individuelles de I_{se} sur S et I_e.

Nous écrirons souvent (S, $\mathcal{F}$), pour désigner l'un ou l'autre des modèles. Cette notation nous permet d'entrer dans le moule (S, *pref*(S)), les individus acteurs étant sous entendus. $\mathcal{F}$ est appelé profil enrichi. Aucune hypothèse n'est faite sur la représentation adoptée pour S et $\mathcal{F}$, celle-ci pouvant être implicite ou explicite. Le **problème de caractérisation des solutions de meilleur compromis** s'énonce alors comme suit :

Étant donnés un ensemble de solutions S en extension ou non et un profil enrichi $\mathcal{F}$, nous cherchons à identifier un plus petit sous-ensemble $C(S, \mathcal{F})$ de S, contenant uniquement les éléments de meilleur compromis pour le profil $\mathcal{F}$. $C(S, \mathcal{F})$ est appelé **ensemble de choix collectif** du profil $\mathcal{F}$ sur S. Comment utiliser au mieux l'information du profil enrichi, pour identifier $C(S, \mathcal{F})$?

Tout d'abord, quand pouvons-nous dire que des individus sont en désaccord ? Si une même solution est choisie individuellement par tous les individus émetteurs du profil, alors cette solution peut être considérée comme une solution choisie collectivement. Si une telle solution n'existe pas, les individus sont dits en conflit global. Plus formellement, notons $C_i(S, \mathcal{F})$ les solutions de S choisies par l'individu i selon ses préférences dans $\mathcal{F}$ et son système de valeurs pour déduire ces solutions[99]. Alors les individus émetteurs de $\mathcal{F}$ sont en **conflit global**

[99] Par 'système de valeurs' nous entendons l'axiomatique de choix utilisée par i pour identifier les solutions qu'il préfère. L'axiomatique de choix se réfère ici clairement aux travaux présentés au § 3.4.

si : $\bigcap_i C_i(S, \mathcal{F}) = \varnothing$. Dans le cas contraire, il suffit de prendre $C(S, \mathcal{F}) = \bigcap_i C_i(S, \mathcal{F})$.

Remarquons que l'absence de conflit global n'empêche pas des désaccords, des divergences d'opinions, des conflits locaux (§ 4.1.1.2). Réciproquement, lorsque le choix local est binaire (utilisation d'une fonction de préférence), tout conflit global se manifeste par des conflits locaux.

En cas de conflit global, comment tirer parti des préférences individuelles pour identifier les solutions de meilleur compromis ?

4.2.1.2. Règles de Décision Collective, d'Agrégation et de Choix Collectif

❖ Règle de Choix Collectif

La façon la plus directe d'obtenir un ensemble de choix est d'utiliser une règle de choix collectif, dont la définition est :

Une **règle de choix collectif** *RCC* est une application qui, à tout ensemble S non vide et tout profil $\mathcal{F}$, associe un ensemble de choix $C(S, \mathcal{F})$.

La règle de choix collectif[100] *RCC* est dite **simple** si $\mathcal{F}$ est un profil simple d'individus I_e sur S, enrichi d'éventuelles informations interpersonnelles. *RCC* est qualifiée de **type sympathie étendue** si $\mathcal{F}$ est un profil type sympathie étendue d'individus émetteurs I_{se} sur S et I_e, enrichi éventuellement d'autres informations interpersonnelles.

Dans la pratique, la règle de choix collectif a été très peu utilisée à cause des difficultés rencontrées pour caractériser directement un ensemble de choix collectif sans passer par des intermédiaires (diviser pour régner). Ces intermédiaires prennent généralement la forme de règles d'agrégation. Présentons-les.

❖ Agrégation, Règle d'Agrégation et Règle de Décision Collective

De manière générale, l'agrégation est la synthèse de l'information afin d'avoir une représentation plus compacte. L'information devient alors plus adaptée au traitement et convient mieux à la perception humaine. Par conséquent, tout processus de description des données en termes d'indicateurs ou de caractéristiques générales, utilise directement ou indirectement des techniques d'agrégation (OTHMANI [1998]).

L'agrégation des préférences correspond donc à la synthèse de l'information préférentielle. Il en existe deux types selon la forme du profil[101] :

Une **règle d'agrégation simple** est une relation fonctionnelle *RA* associant à tout profil $\mathcal{F}$ d'individus I_e sur un ensemble S, les préférences $pref(I_e, S)$ de l'ensemble I_e sur S :

$$pref(I_e, S) = RA(\mathcal{F}(I_e, S)) \ . \qquad (4\text{–}16)$$

Une **règle d'agrégation étendue** est une relation fonctionnelle *RA* associant à tout profil type sympathie étendue $\mathcal{F}$ d'individus émetteurs I_{se} sur un ensemble de solutions potentielles S et de concernés I_e, les préférences $pref(I_{se}, S)$ de I_{se} sur S :

$$pref(I_{se}, S) = RA(\mathcal{F}(I_{se}, I_e, S)) \ . \qquad (4\text{–}17)$$

---

[100] La règle de choix collectif a été beaucoup utilisée en théorie des jeux coopératifs (MOULIN [1988], [1995]), sous le nom de **règles d'allocation**. Pour une comparaison de cette théorie et de celle du choix social, voir MOULIN & THOMSON [1997], SEN [1986 § 5] parmi d'autres.

[101] SEN [1986 § 7.1] classe les problèmes d'agrégation en deux catégories : l''*interest-aggregation*' et le '*judgment-aggregation*', correspondant respectivement à l'agrégation étendue et l'agrégation simple.

Les préférences $pref(I_e, S)$ et $pref(I_{se}, S)$ sont appelées **préférences collectives** de la règle d'agrégation RA. Elles sont notées respectivement $pref_{RA(\mathcal{F})}(I_e, S)$ et $pref_{RA(\mathcal{F})}(I_{se}, S)$ pour notifier qu'elles sont issues de la transformation du profil $\mathcal{F}$ par la règle d'agrégation RA.

Nous nous intéresserons plus particulièrement aux règles d'agrégation aboutissant à des préférences collectives structurées en relation binaire de préférence (Figure 21). Ces règles d'agrégation se ramènent à une règle de choix collectif, en transformant l'instance initiale $(S, I_e, \mathcal{F})$ (resp. $(S, I_e, I_{se}, \mathcal{F})$) en une instance $(S, \succcurlyeq)$, puis en utilisant un algorithme recherchant un ensemble de choix $C_{SA}(S, \succcurlyeq)$ de S identifié par un système axiomatique SA donné (§ 3.4).

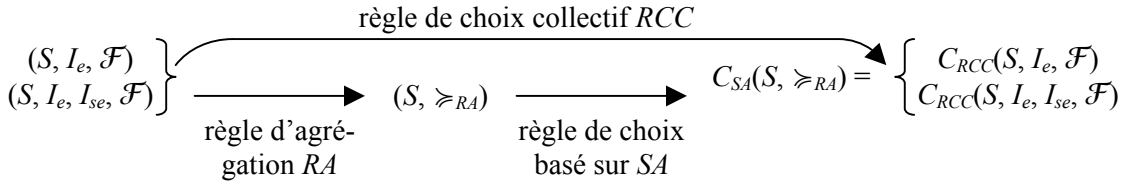


Figure 21 : Relations exploitées entre les règles de décision, les ensembles de choix et les profils, pour l'aide au choix.

Remarque : la relation $\succcurlyeq_{RA(\mathcal{F})}$ est l'image par RA du profil $\mathcal{F}$. Mais pour ne pas alourdir la notation et lorsque le profil concerné par l'application est clairement identifié par le contexte, nous noterons simplement $\succcurlyeq_{RA}$.

Cette manière de découper le problème d'identification d'un ensemble de choix collectif, en deux problèmes partiels séquentiels, est une possibilité parmi d'autres. Nous en reparlerons au § 4.2.1.5.

Remarquons que toute règle de décision simple, donc manipulant des préférences individuelles de S, peut s'employer sur un profil type sympathie étendue $\mathcal{F}(I_{se}, I_e, S)$ d'au moins deux manières différentes :

– En n'utilisant que les jugements communs aux deux types de profils selon la relation **(**4-14**)**. Cette relation se traduit pour les préférences individuelles binaires par :

$$\forall\ (x, y, i) \in S \times S \times I_e, \quad p_i(x, y) = p_i((x, i), (y, i)) \qquad (4\text{–}18)$$

avec p_i la fonction de préférence de l'individu i sur S, et aussi – par abus d'écriture – la fonction de préférence de l'individu i sur les positions de $S \times I_e$.

Les préférences collectives sont alors des jugements collectifs sur S : $pref(I_{se}, S)$.

– En considérant le profil $\mathcal{F}(I_{se}, I_e, S)$ comme étant le profil simple $\mathcal{F}(I_{se}, S \times I_e)$, établissant ainsi des préférences collectives sur $S \times I_e$: $pref(I_{se}, S \times I_e)$.

Nous utilisons ici le terme de **règle de décision collective** pour désigner de manière générique la règle de choix collectif et la règle d'agrégation.

4.2.1.3. Exemples de Règles de Décision Collective

Toute transformation d'un profil en des préférences de base ou en un sous-ensemble de solutions est une règle de décision collective. Différentes disciplines en ont utilisé. Nous en présentons quelques unes.

❖ Les Procédures de Vote

Les procédures de vote portent exclusivement sur les ordres totaux individuels. Nous en présentons deux qui sont la règle de Borda et la règle de la majorité. Ces deux règles ont une très grande valeur historique et ont fait l'objet d'un grand nombre de travaux théoriques. Pour un approfondissement sur les procédures de vote, nous conseillons l'article MARCHANT *et al.* [2000].

La **règle de Borda**[102] est une règle d'agrégation d'un profil d'ordres totaux individuels $\succcurlyeq_i$ sur un ensemble <u>fini</u> S, dont la préférence collective associée $\succcurlyeq_{\mathrm{B}}$ est le préordre total sur S défini par :

$$\text{Pour } x, y \in S : \quad x \succcurlyeq_{\mathrm{B}} y \Leftrightarrow \sum_{z \in S} \mathrm{card}(CO(z \succcurlyeq x)) \leq \sum_{z \in S} \mathrm{card}(CO(z \succcurlyeq y)) \ .$$

$CO(x \succcurlyeq y)$ étant la coalition d'individus jugeant x au moins aussi bon que y. Il existe plusieurs définitions équivalentes de la règle de Borda. La définition présentée a l'avantage de pouvoir se généraliser aux profils de préordres totaux individuels, voire à tout profil de relations binaires de préférence individuelles sans modifier la formule. Cette généralisation est appelée ici, **règle de Borda généralisée**. Dans sa thèse, MARCHANT élargit la règle de Borda aux relations binaires de préférence floues (§ 3.3.2.2). D'autres extensions (MARCHANT *et al*. [2000], DURAND [2000]) ont vu le jour avec les méthodes de scorage ou encore la **fonction de choix de Borda**. Cette dernière associe à tout profil d'ordres totaux une relation non-binaire de préférence $rnp_{\mathrm{B}}(S) = (\mathcal{P}(S) \setminus \{\varnothing\}, C_{\mathrm{B}})$ vérifiant que pour tout sous-ensemble non vide A de S,

$$C_{\mathrm{B}}(A) = \{x \in A \text{ tel que : } \mathrm{rang}(x, A) = \min_{y \in A} \{\mathrm{rang}(y, A)\} \},$$

$$\text{avec } \mathrm{rang}(z, A) = \sum_{w \in A} \mathrm{card}(CO(w \succcurlyeq z)) \text{ pour tout } z \in A.$$

La **règle de la majorité**[103] (ou **méthode majoritaire**) est une règle d'agrégation d'un profil d'ordres totaux individuels $\succcurlyeq_i$ sur un ensemble S, dont la préférence collective $\succcurlyeq_{\mathrm{M}}$ est une relation binaire de préférence définie par :

$$\text{Pour } x, y \text{ de } S, \quad x \succcurlyeq_{\mathrm{M}} y \Leftrightarrow \mathrm{card}(CO(x \succcurlyeq y)) \geq \mathrm{card}(CO(y \succcurlyeq x)) \ .$$

$CO(x \succcurlyeq y)$ étant la coalition d'individus jugeant x au moins aussi bon que y. Cette préférence collective ne conserve pas la transitivité. La règle de choix collectif associée choisit une seule solution de compromis, si elle existe, et est majoritairement meilleure que toutes les autres. Elle est dite **vainqueur de Condorcet**.

La règle de la majorité peut se généraliser – en conservant la même formule – aux préférences individuelles structurées en préordres totaux (SEN [1970]) ou en tournoi, voire à toute relation binaire de préférence. Cette version est ici appelée **règle de la majorité généralisée**. Des généralisations sont apparues comme la proportionnelle ou encore la règle de concordance élémentaire sur laquelle est basée la règle d'agrégation ELECTRE I (ROY & BOUYSSOU [1993], SCHÄRLIG [1996]).

Remarquons que la règle de la majorité suggère aussi des règles valuées, comme par exemple la suivante[104] : Pour $x, y \in S$ et $\alpha \in \mathcal{PR}(AF(RBP))$, $\mu_\alpha(x, y) = \mathrm{card}(CO(x \ \alpha \ y))$.

Remarquons également que la règle de la majorité est équivalente à une règle de Borda mais limitée aux seuls couples comparés.

❖ Utilitarisme

Avec Borda et Condorcet, c'est l'une des plus ancienne règle qui s'est développée non plus en politique, mais en économie avec BENTHAM. Jusqu'à ARROW [1951], les procédures de vote étaient étudiées comme des curiosités, alors que l'utilitarisme a connu un essor très im-

[102] Cette règle d'agrégation a été définie par Jean-Charles de BORDA en 1781, dans son '*Mémoire sur les Élections au Scrutin*', Mémoire de l'Académie des sciences, Paris.

[103] Cette règle d'agrégation a été définie par Mary Jean-Antoine Caritat Marquis de CONDORCET en 1785 dans son '*Essai sur l'Application de l'Analyse à la Probabilité des Décisions Rendues à la Pluralité des Voix*', Imprimerie Royale, Paris.

[104] Voir par exemple KÖHLER [1978] et ARROW & RAYNAUD [1986].

portant (fondements de l'économie théorique classique) dès son apparition notamment avec son application dans le système carcéral anglais par BENTHAM[105].

La **règle d'utilité additive** est une règle d'agrégation sur un profil simple de vrai-critères $g_i = (f_i, \geq)$ sur S, dont les préférences collectives sont structurées en vrai-critère $g_{\mathrm{UA}} = (f_{\mathrm{UA}}, \geq)$ défini comme suit :

$$\text{Pour tout } x \in S : \ f_{\mathrm{UA}}(x) = \sum_{i \in I_e} u_i(f_i(x))$$

où u_i est une fonction réelle de la variable réelle. Plus généralement, toute fonction analytique réelle de $\mathrm{card}(I_e)$ variables réelles peut faire office de règle d'agrégation.

❖ Règles Standards Générales

Les règles de Pareto et Lexicographique sont deux règles très utilisées en théorie du choix social et leurs versions limitées aux profils de vrai-critères sont très utilisées dans les applications pratiques multicritères (MOP) de RO (EHRGOTT & GANDIBLEUX [2000]) et en économie.

La **règle de Pareto**[106] P est une règle d'agrégation qui, à tout profil simple de relations binaires de préférence $\succcurlyeq_i$ sur un ensemble S, associe la relation binaire de préférence $\succcurlyeq_{\mathrm{P}}$ définie par :

$$\text{Pour } x, y \text{ de } S, \qquad x \succ_{\mathrm{P}} y \ \Leftrightarrow \ [\ \forall\, i \in I_e,\ x \succcurlyeq_i y\ ] \ \text{ et } \ [\exists\, i \in I_e \text{ tel que : } x \succ_i y\ ],$$

$$x \simeq_{\mathrm{P}} y \ \Leftrightarrow \ [\ \forall\, i \in I_e,\ x \simeq_i y\ ]\ .$$

La **règle lexicographique** LX est une règle d'agrégation qui, à un profil simple de préordres totaux individuels $\succcurlyeq_i$ sur un ensemble S, et un ordre total sur l'ensemble I_e des individus associe le préordre total collectif $\succcurlyeq_{\mathrm{LX}}$ défini par :

$$\text{Pour } x, y \text{ de } S, \qquad x \succ_{\mathrm{LX}} y \ \Leftrightarrow \ \exists\, i \in I_e \text{ tel que : } [\ x \simeq_k y\ \forall\, k < i\ ] \ \text{ et } \ [\ x \succ_i y\ ],$$

$$x \simeq_{\mathrm{LX}} y \ \Leftrightarrow \ [\ \forall\, i \in I_e,\ x \simeq_i y\ ]\ .$$

❖ Procédures d'Aide Multicritère à la Décision pour une Problématique de Choix

L'AMCD (VINCKE [1989], ROY & BOUYSSOU [1993]) s'intéresse aux profils simples de fonction-critères, enrichis éventuellement d'information inter-critère, qui sont agrégés en relation binaire de préférence ou en relation binaire de préférence floue. Les règles d'agrégation sont appelées **PAMC** (procédures d'agrégation multicritère).

Différents systèmes d'aide à la décision ont été conçus et commercialisés dans le cadre d'une problématique de choix. Par exemple, les systèmes ELECTRE I et IS (ROY & BOUYSSOU [1993], SCHÄRLIG [1996], AÏT YOUNES *et al.* [2000*a*, 2000*b*]) utilisent :

- un profil pondéré de fonction-critères sur un ensemble de solutions alternatives en extension ;
- une relation d'agrégation simple $RA1$ usant des notions de coalitions concordantes et discordantes, retournant une relation de surclassement $\succcurlyeq_{RA1}$ pas nécessairement transitive ;
- le noyau (§ 3.4.3.2) comme système axiomatique pour trouver un ensemble de choix à partir de la relation de surclassement : $C(S, \succcurlyeq_{RA1}) = N(S, \succcurlyeq_{RA1})$. Et lorsqu'il n'en existe pas : $N(S, \succcurlyeq_{RA1}) = \varnothing$, un autre problème est résolu, de manière à ce que la nouvelle relation de surclassement obtenue $\succcurlyeq_{RA2}$ vérifie $N(S, \succcurlyeq_{RA2}) \neq \varnothing$.

Ce type de système illustre bien l'utilisation de la décomposition de la Figure 21 dans un système informatique réel.

---

[105] BENTHAM Jeremy [1789] – *An Introduction to the Principles of Morals and Legislation* – Londres, Payne ; *réimpr.* [1907] Oxford, Clarendon Press.

[106] Cette règle a été introduite par Vilfredo PARETO en 1897 dans son '*Cours d'Economie Politique*', Rouge, Lausanne.

❖ Règles d'Agrégation Étendue

Ce type de règles est de plus en plus utilisé en théorie du choix social et en éthique pour résoudre les problèmes de distribution de ressources entre différents individus d'une population. Nous présentons ici l'une de ces règles d'agrégation.

La **Règle de Rangement Interpersonnelle**[107] IROR **(**= *Interpersonal Rank Order Rule***)** définit une relation binaire de préférence notée $\succcurlyeq_{\text{IROR}}$ sur un ensemble fini de solutions S, à partir du préordre total étendu $\succcurlyeq$ sur le produit cartésien de S et d'un ensemble fini d'individus I_e, comme suit :

Notons $N(\succcurlyeq, x, i)$ l'ensemble des positions de $S \times I_e$ jugées par $\succcurlyeq$ de moins bonne qualité que (x, i) : $N(\succcurlyeq, x, i) = \{(y, j) \in S \times I_e$ tel que : $(x, i) \succcurlyeq (y, j)\}$. Alors,

$$\text{pour } x, y \text{ de } S : \quad x \succcurlyeq_{\text{IROR}} y \iff \sum_{i \in I_e} \text{card}(N(\succcurlyeq, x, i)) \geq \sum_{i \in I_e} \text{card}(N(\succcurlyeq, y, j))$$

IROR est la version interpersonnelle de la règle de Borda. De même, il est possible de définir une version interpersonnelle de la règle de la majorité en limitant $N(\succcurlyeq, x, i)$ aux seuls éléments de $\{x, y\} \times I_e$. Une généralisation de IROR peut être trouvée dans SEN [1977*a*].

D'autres règles d'agrégation étendues seront présentées au chapitre 6.

4.2.1.4. Quelques Propriétés Fondamentales

Nous présentons ici deux types de propriétés fondamentales liées aux règles d'agrégation.

❖ Conservation de Propriétés Structurelles de l'Individuel au Collectif

Comme les relations binaires de préférence occupent une place privilégiée dans ce document, seules les formes de conservation les concernant sont présentées.

Considérons une règle d'agrégation *RA* sur un profil simple ou de type sympathie étendue $\mathcal{F}$ formé de relations binaires de préférence, dont les préférences collectives sont structurées en relation binaire de préférence $\succcurlyeq_{RA}$ sur S. Alors :

RA **conserve la transitivité** ssi : $[\forall \succcurlyeq \in \mathcal{F}, \succcurlyeq$ est transitive$] \Rightarrow [\succcurlyeq_{RA}$ est transitive$]$

RA **conserve la P-acyclicité** ssi : $[\forall \succcurlyeq \in \mathcal{F}, \succcurlyeq$ est P-acyclique$] \Rightarrow [\succcurlyeq_{RA}$ est P-acyclique$]$

La conservation peut se généraliser à toute propriété de la relation binaire de préférence.

Par exemple la règle de Pareto conserve la transitivité, mais ne conserve pas la P-acyclicité comme le montre la Figure 22.

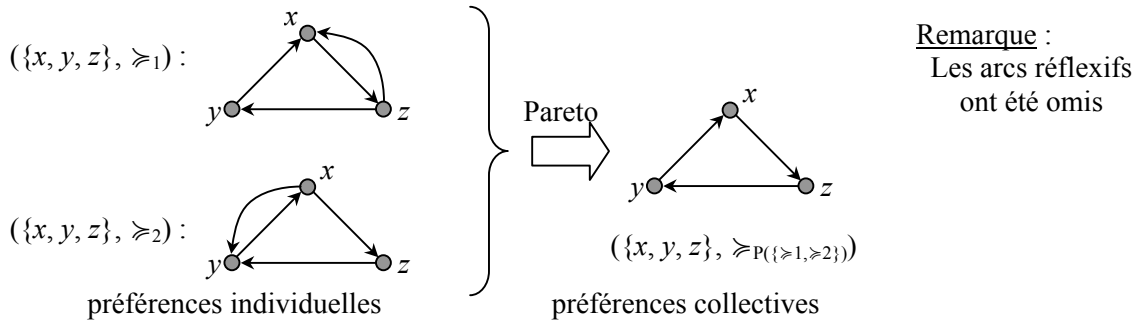


Figure 22 : Exemple de la non-conservation de la P-acyclicité par la règle de Pareto.

Le tableau suivant récapitule quelques propriétés de conservation de la règle de Pareto :

[107] Cette règle d'agrégation a été définie dans SEN [1977*a*].

Propriétés des préférences individuelles	Propriétés des préférences collectives
Préordres partiels	Préordre partiel
Relations P-acycliques	Quelconques à partir de deux individus et 3 solutions
Relations quasi-transitives	À partir de 3 solutions : peut comporter des cycles, sauf pour 2 individus et 3 solutions (P-acyclique).

Tableau 6 : Quelques propriétés structurelles de la règle de Pareto.

Des investigations combinatoires plus poussées peuvent être menées pour connaître les cas où une propriété est conservée.

Un autre exemple : La règle de la majorité ne conserve ni la transitivité, ni la quasi-transitivité, ni la P-acyclicité dans le cas général. Cette règle a été très étudiée. Pour des investigations sur la conservation de la transitivité pour la règle de la majorité, voir SEN [1970], ARROW & RAYNAUD [2000], RAYNAUD [2000].

Le terme de **dominance** est utilisé pour désigner les règles d'agrégations dont les préférences collectives sont structurées en préordre, et dont les préférences individuelles sont structurées en vrai-critères (simples ou étendues). Par exemple la règle de Pareto portant sur un profil simple de vrai-critères, est appelée **dominance de Pareto**, car elle conserve la transitivité. D'autres exemples seront donnés au chapitre 6.

Par abus de langage, nous dirons qu'une règle d'agrégation est une extension d'une autre règle d'agrégation, lorsque les préférences collectives de la première sont une extension des préférences collectives de la seconde (§ 3.3.1.1). Il en est de même pour l'inclusion. Rappelons que ces relations sont transitives. De plus, une règle d'agrégation est qualifiée de partielle, si ses préférences collectives sont structurées en relation binaire de préférence admettant l'incomparabilité.

❖ Mesurabilité et Commensurabilité

Nous avons vu aux § 3.3.3.6 et 4.1.3.4 que les décisions pouvaient être classées selon deux types d'axiomes d'invariance, respectivement les axiomes de mesurabilité et les axiomes de commensurabilité. Cette classification peut aussi être utilisée pour classer les règles de décision collective sur des profils de relations binaires de préférence.

Pour cela, il est nécessaire de définir la relation d'inclusion d'information préférentielle sur l'ensemble des ensembles de profils. Par exemple, un vrai critère contient toute l'information préférentielle véhiculée par le préordre total. On peut découper l'ensemble des vrai-critères en classes d'équivalence, chaque classe ayant le même préordre total sous-jacent. Si une règle de décision collective est applicable sur les profils de préordres totaux, il est aussi applicable sur les profils de vrai-critères. Elle vérifie dans ce cas l'axiome de mesurabilité ordinale. Nous renvoyons à GUITOUNI *et al*. [2000] en plus des références aux § 3.3.3.6 et 4.1.3.4 pour cette forme de classification.

4.2.1.5. Alternatives à la Règle d'Agrégation

Choisir de modéliser les préférences individuelles c'est déjà faire le choix d'utiliser l'ordinateur à d'autres fins que celles du support communicatif et combinatoire. La construction de règles de choix collectif est une première alternative à la règle d'agrégation. En voici d'autres :

❖ Interactivité Pure

Comme nous l'avons vu précédemment, il est possible d'utiliser des systèmes qui ne font usage d'aucune représentation des préférences individuelles. Par exemple, ceux basés sur l'interaction directe avec un (planificateur, coordinateur) ou plusieurs acteurs (auto-planification). Mais nous avons vu leurs inconvénients. Ils demandent notamment des efforts

cognitifs importants aux utilisateurs. Face à l'ampleur de la tâche, des effets de fatigue peuvent influer sur le choix, etc.. Nombreux sont les avantages par rapport à la résolution à la main. En effet, si l'outil logiciel est bien conçu, il décuple considérablement les possibilités de l'utilisateur. Il développe le sens des responsabilités et de la négociation de chaque acteur (dans le cadre de décision en groupe). Et il donne l'impression d'un meilleur contrôle par rapport aux systèmes basés sur les règles d'agrégation ne faisant usage que d'une fonction objective collective unique (Utilitarisme).

La modélisation des préférences et l'étude des règles d'agrégation peuvent contribuer à améliorer de tels systèmes. Notamment elles permettront de proposer à l'utilisateur des contraintes temporaires plus adéquates, modélisant les préférences individuelles ou basées sur un concept particulier agrégatif, comme l'équité ou/et l'efficacité. Ce dernier point permettra aux utilisateurs de poser des contraintes temporaires du type : « supprimer les valeurs de domaines qui impliquent uniquement des plannings moins équitables qu'un planning de référence ».

❖ Hiérarchie Agrégative

Une autre possibilité est d'utiliser une **hiérarchie agrégative**. En d'autres termes une agrégation comportant plusieurs niveaux. Par exemple, lorsque le profil est formé des préférences de I_e acteurs, chaque acteur i ayant $n(i)$ critères individuels, les préférences globales peuvent être obtenues par un raisonnement agrégatif se décomposant en deux niveaux : Une première agrégation au niveau de chaque décideur, suivie d'une seconde agrégation collective. La hiérarchie agrégative permet d'appliquer le principe de « diviser pour régner » pour élaborer une règle d'agrégation.

Cette méthodologie est généralement la règle dans les problèmes complexes, et plusieurs méthodologies de conception ont été développées (KEENEY & RAÏFFA [1976], KEENEY [1988], ROY [1985], CORNER *et al*. [2001]).

Les hiérarchies agrégatives peuvent mixer les deux formes d'agrégation : simple et étendue. Les règles d'agrégation sur les profils simples peuvent être utilisées sur les profils type sympathie étendue pour obtenir un profil étendu, ou un autre profil de type sympathie étendue. Par exemple, la règle de Pareto P peut être utilisée sur profil étendu de relations binaires de préférence $\succcurlyeq_i$ sur un ensemble $S \times I_e$, pour obtenir une relation binaire de préférence $\succcurlyeq_P$ sur $S \times I_e$.

Un graphe orienté, appelé **graphe d'agrégation**, peut être associé à une hiérarchie agrégative. Les sommets correspondent aux différentes préférences individuelles et collectives qui intervenent dans la hiérarchie. Un couple de sommets ($pref_1$, $pref_2$) est un arc du graphe s'il existe une règle d'agrégation RA telle que $pref_1$ forme les préférences d'un individu du profil agrégé par RA, et $pref_2$ les préférences collectives. Le graphe d'agrégation est acyclique : Les préférences identifiées en un sommet peuvent être utilisées à différents niveaux.

Remarquons que la hiérarchie agrégative permet en partie d'étudier certains processus de décision. L'évolution des préférences des utilisateurs est mesurée en comparant les écarts entre la solution du processus de décision et les solutions de la hiérarchie agrégative. De nouvelles règles peuvent être aussi construites en utilisant l'apprentissage.

❖ Systèmes Multi-Agents

Ce type de méthode consiste à utiliser des agents logiciels, à raison d'un agent par individu du profil, qui simulent le travail de négociation. Chaque agent fait valoir les préférences de l'individu associé lors d'interactions, et le système possède certaines propriétés pour faire converger la négociation. La négociation simulée permet d'aboutir à une solution qui est prise comme solution de meilleur compromis. Ces systèmes sont très récents et permettent entre autre d'étudier les propriétés émergeantes d'interactions entre plusieurs agents (FERBER [1995]).

Utilisés de cette façon, les systèmes multi-agents sont assimilables à une méta-heuristique au même titre que la recherche tabou, le recuit simulé ou les algorithmes génétiques.

❖ Autres Approches

De nouvelles approches ont fait leur apparition ces dernières années. Citons l'approche des ensembles incertains basés sur la dominance (*Dominance-Based Rough Set Approach* : GRECO *et al*. [2001]).

Des procédures mixtes sont aussi possibles comme l'utilisation de la dominance de Pareto couplée à la méthode interactive. Chaque procédure a ses avantages et inconvénients. Les systèmes mixtes permettent de mettre en avant les avantages de chaque méthode.

Rien n'empêche de découper le problème général (choix collectif) différemment, ou d'utiliser les problèmes partiels autrement.

Le système ELECTRE IS illustre bien ce dernier cas : il utilise le noyau pour identifier les solutions de meilleur compromis aux solutions de meilleure qualité de la relation de surclassement intermédiaire (voir § 4.2.1.3). Sachant que le noyau n'identifie pas toujours un sous-ensemble non vide, l'agrégation suivie du choix via la relation de surclassement est complétée par un autre problème partiel modifiant la relation de surclassement $\succcurlyeq_{RA}$ pour qu'elle soit adaptée au noyau.

JASZKIEWICZ [2001], lui, utilise une itération de trois problèmes partiels : une agrégation (Pareto), un choix (approximation et échantillonnage de l'ensemble maximal) et une procédure de tri (§ 3.2.3.3).

4.2.2. Choisir une Règle de Décision Collective Adéquate par Analyse Axiomatique

Tout d'abord, pourquoi est-il nécessaire de choisir ?

Comme nous avons vu avec les exemples du § 4.2.1.3, il est très facile de créer une règle d'agrégation. Par conséquent n'importe quelle règle peut être utilisée. Dans la pratique, une prolifération de règles intuitives, plus ou moins scientifiques ou critiquables, ont vu le jour. Chaque concepteur propose sa "recette" particulière (RAYNAUD [2000]). Pourtant chaque règle identifie une décision (préférences collectives, ensemble de choix collectif) souvent différente.

Comment identifier les règles les mieux adaptées à notre application ?

La théorie du choix social a proposé une approche donnant des fondements aux règles de décision collective : l'analyse axiomatique.

4.2.2.1. Définition de l'Analyse Axiomatique pour la Décision Collective

L'analyse axiomatique[108] est un ensemble de méthodes qui permettent de formaliser le problème de choix de règles de décision collectives adéquates, en se basant sur des axiomes. Lorsqu'elle est bien menée, elle permet entre autre de :

- mettre de la clarté dans des règles souvent opaques et complexes ;
- justifier l'utilisation de certaines règles ;
- étudier les liens entre différentes règles ;
- identifier des propriétés (axiomes) communes aux règles ;
- explorer la compatibilité de propriétés pris conjointement ; et
- observer leurs comportements, leur acceptabilité sur des cas réels.

En termes géométriques, un axiome identifie un sous-ensemble de règles de décision collectives. Il est aussi appelé propriété ou encore principe lorsqu'il véhicule un sens, une idée,

[108] Pour des documents portant sur l'analyse axiomatique nous renvoyons à SEN [1970], FISHBURN [1987], PERNY [1992], THOMSON [1997*a*], [1997*b*].

un principe clair et distinct. Les axiomes permettent de mettre en œuvre le principe « diviser pour régner » : Chaque axiome véhicule une information indépendante du point de vue de la logique et des concepts (THOMSON [1997*a*]), le tout caractérisant la ou les règles étudiées[109]. Les axiomes sont formulés pour contraindre les différents objets intervenants dans une décision collective usant de préférences. Ils peuvent ainsi être classés selon qu'ils portent sur :

- les degrés de liberté accordés aux individus dans leurs préférences,
- la structure des buts collectifs recherchés, ou
- les relations entre les jugements collectifs et les choix individuels.

Notons *RDC* la règle de décision collective recherchée.

Les axiomes du premier point expriment une restriction sur l'ensemble de définition de *RDC*. Par exemple, les profils doivent être des ensembles de préordres partiels, ou encore, les profils doivent être de type sympathie étendue satisfaisant l'axiome d'identité, le domaine des solutions *S* a une forme particulière, certaines préférences individuelles d'un groupe d'émetteurs sont corrélées, etc..

Les axiomes du second type manifestent une restriction de l'ensemble image de *RDC*. Par exemple, les préférences collectives doivent être transitives, ou admettre un ensemble optimal non vide, etc..

Le dernier type d'axiomes formulent des restrictions sur les correspondances possibles pour *RDC* entre l'ensemble des profils et l'ensemble des préférences collectives. Par exemple, l'influence d'un ensemble de solutions *S* instable sur *RDC*, le respect de l'unanimité ou d'une forme d'équité, etc.

4.2.2.2. Quelques Exemples d'Axiomes

Nous détaillons ici six axiomes qui nous serons utiles pour la suite : les deux premiers concernent respectivement le domaine de définition et l'ensemble d'arrivée des règles de décision collectives. Les quatre suivants sont des propriétés pertinentes des règles d'agrégation : l'Unanimité, l'axiome de Pareto, l'axiome IIPC et l'axiome IIIC.

Axiome DOM(POP) (Domaine non Restreint des Préférences Individuelles structurées en Préordres Partiels) : Une règle de décision collective *RDC* vérifie l'axiome DOM(POP) si son domaine de définition est restreint à l'ensemble des profils de préordres partiels ; c'est à dire $\mathcal{P}$(POP, *S*) dans le cas simple, et $\mathcal{PSE}$(POP, *S*, I_e), $\mathcal{PE}$(POP, *S*, I_e) ou $\mathcal{PG}$(POP, *S*, I_e) dans le cas étendu.

Cet axiome limite les différents profils pouvant être traité par la règle *RDC* aux profils de préordres partiels. Il peut être généralisé à tout sous-ensemble de relations binaires de préférence. Nous pouvons même l'utiliser pour tout type de préférences binaires PREF : DOM(PREF).

Axiome IM(POP) (Préférences Collectives Restreintes au Préordre Partiel) : Une règle d'agrégation *RA* dont les préférences collectives $\succcurlyeq_{RA}$ sont structurées en relation binaire de préférence vérifie l'axiome IM(POP) si, l'image par *RA* de tout profil $\mathcal{F}$ du domaine de définition de *RA* est structuré en préordre partiel ($\Leftrightarrow \succcurlyeq_{RA(\mathcal{F})}$ est réflexive et transitive).

Cet axiome limite les préférences collectives aux préordres partiels sur *S*. Remarquons qu'une règle d'agrégation qui conserve la transitivité vérifie conjointement l'axiome DOM(POP) et IM(POP). C'est le cas de la dominance de Pareto, contrairement à la règle de la majorité.

Axiome d'Unanimité U[α] : Une règle d'agrégation simple *RA*, dont les préférences collectives et préférences individuelles sont basées sur le même ensemble d'attitudes fondamen-

[109] Remarquons l'analogie avec les problèmes de satisfaction de contraintes : les axiomes sont des contraintes qui caractérisent les règles de décision collectives qui sont les solutions réalisables.

tales AF, satisfait l'axiome d'unanimité U[α] pour l'attitude $\alpha \in \mathcal{PR}(AF)$ si, quelque soit $\mathcal{F}(I_e, S) = \{\succcurlyeq_i\}_i$, l'implication suivante est vérifiée :

Pour tout $x, y \in S$: [pour tout $i, j \in I_e$, $p_i(x, y) = p_j(x, y) = \alpha$] $\Rightarrow$ $p_{RA}(x, y) = \alpha$.

En d'autres termes, RA respecte l'unanimité sur les préférences par paires.

Axiome (d'agrégation) de Pareto (ou *strong Pareto principle*) : Une règle d'agrégation simple RA, dont :

(a) le domaine de définition est l'ensemble des profils simples $\mathcal{P}(\mathsf{RBP}, S)$ de relations binaires de préférence, et

(b) les préférences collectives sont structurées en relation binaire de préférence,

satisfait l'axiome de Pareto si pour tout $\mathcal{F}(I_e, S) = \{\succcurlyeq_i\}_i$, l'implication suivante est vérifiée : Pour tout $x, y \in S$: [pour tout $i \in I_e$, $x \succcurlyeq_i y$ et il existe $i \in I_e$, $x \succ_i y$] $\Rightarrow$ $x \succ_{RA} y$.

L'axiome de Pareto véhicule un **principe d'efficacité** (par rapport à l'unanimité) dans le sens où si tout le monde est unanime pour dire que x est au moins aussi bon (ou surclasse) que y et qu'il existe une personne préférant strictement x à y, alors x est collectivement préféré à y. Les deux axiomes précédents sont vérifiés par différents exemples de règles présentées (Borda, majorité, Pareto, lexicographique, PAMC, etc.). La règle de Pareto est caractérisée à la fois par l'axiome de Pareto, qui définit sa partie asymétrique, et l'axiome d'unanimité en α = indifférent, qui décrit sa partie symétrique. Toute règle satisfaisant ces deux axiomes est une extension de la règle de Pareto.

Axiome d'indépendance aux comparaisons personnelles non pertinentes IIPC (*Independence of Irrelevant Personal Comparisons*) : Considérons deux profils $\mathcal{F}_1(I_e, S) = \{p_{1,i}\}_i$ et $\mathcal{F}_2(I_e, S) = \{p_{2,i}\}_i$ de préférences binaires sur S issus du domaine de définition d'une règle d'agrégation simple RA. Les profils sont décrits par leurs fonctions de préférence individuelles. Alors, RA vérifie l'axiome IIPC si pour tout couple de solutions x, y de S, nous avons :

$$[\text{pour tout } i \in I_e,\ p_{1,i}(x, y) = p_{2,i}(x, y)] \Rightarrow p_{RA(1)}(x, y) = p_{RA(2)}(x, y), \qquad (4\text{–}19)$$

avec $p_{RA(k)}$, pour $k \in \{1, 2\}$, la fonction de préférence des préférences collectives de RA lorsqu'elle est appliquée au profil $\mathcal{F}_k(I_e, S)$.

Cet axiome impose que, tant que les préférences individuelles restent inchangées sur un couple de solutions de S, la règle d'agrégation émet le même jugement sur ce couple[110]. Le jugement sur un couple de solutions n'est pas influencé par les préférences individuelles sur d'autres couples.

Une généralisation aux règles d'agrégation admettant des préférences collectives structurées en relation non binaire de préférence a été formulée (voir par exemple SEN [1986 § 1.3]) : tant que les préférences individuelles restent inchangées sur un sous-ensemble de S, l'ensemble de choix associé à ce sous-ensemble pour chaque profil, reste aussi inchangé.

Cet axiome est applicable aux règles d'agrégation étendues si on utilise la relation (4–18) de la page 135. La formule (4–19) ne contraint alors que les couples (x, i) et (y, i). Par conséquent, l'axiome IIPC impose aussi une indépendance vis-à-vis des comparaisons interpersonnelles faisant intervenir à la fois les solutions x et y, et les solutions de $S \setminus \{x, y\}$. L'axiome

[110] Cet axiome est plus connu sous le nom d'axiome d'indépendance aux solutions alternatives non pertinentes (*Independance of Irrelevant Alternatives*). Mais comme le faisait remarquer HAMMOND [1991], un tel nom est ambigu car, cet axiome impose une indépendance aux préférences de chaque individu, et non une indépendance par rapport à la présence ou l'absence de solution, c'est-à-dire une indépendance vis-à-vis d'un ensemble de solutions S variable. Un tel axiome existe et est connu sous le nom de menu-indépendance (voir à ce propos SEN [1993] et [1997]).

suivant impose une indépendance des comparaisons interpersonnelles uniquement en dehors du couple de solutions (x, y) :

Axiome d'indépendance aux comparaisons interpersonnelles non pertinentes IIIC (*Independence of Irrelevant Interpersonal Comparisons*) : Considérons deux profils $\mathcal{F}_1(I_{se}, I_e, S) = \{p_{1,\,i}\}_i$ et $\mathcal{F}_2(I_{se}, I_e, S) = \{p_{2,\,i}\}_i$ de préférences binaires étendues sur $S \times I_e$ issus du domaine de définition d'une règle d'agrégation étendue *RA*. Les préférences individuelles des profils sont décrites par leurs fonctions de préférence. Alors, *RA* vérifie l'axiome IIPC si pour tout couple de solutions x, y de S, nous avons :

$$\left[\begin{array}{c}\text{pour tout } (z,w,i,j,k) \in \{x,y\} \times \{x,y\} \times I_e \times I_e \times I_{se}, \\ p_{1,k}((z,i),(w,j)) = p_{2,k}((z,i),(w,j))\end{array}\right] \Rightarrow p_{RA(1)}(x, y) = p_{RA(2)}(x, y),$$

avec $p_{RA(k)}$, pour $k \in \{1, 2\}$, la fonction de préférence des préférences collectives de *RA* lorsqu'elle est appliquée au profil $\mathcal{F}_k(I_{se}, I_e, S)$.

Par rapport à l'axiome IIPC, cet axiome impose uniquement une indépendance des préférences collectives vis-à-vis des préférences individuelles sur d'autres couples que x, y. Une généralisation aux règles d'agrégation admettant des préférences collectives structurées en relation non binaire de préférence a été formulée (voir par exemple ROBERTS [1979]). Ni la règle de Borda, ni IROR ne satisfont les axiomes IIPC et IIIC. Par contre la version interpersonnelle de la règle de la majorité (fin du § 4.2.1.3 en page 138) vérifie IIIC mais pas IIPC. Toute règle satisfaisant IIPC satisfait IIIC. Mais la réciproque est fausse.

Nous présenterons d'autres axiomes dans la suite de ce document.

4.2.2.3. Analyse Axiomatique, Hiérarchie Agrégative et Processus de Décision

Les hiérarchies agrégatives, comme les processus de décision, possèdent un inconvénient majeur : Elle est très manipulable. Prenons l'exemple de la règle de la majorité. Utilisée dans une procédure itérative, elle peut amener à ne pas choisir le vainqueur de Condorcet. C'est le cas de la technique des **amendements** stratégiques utilisée par les assemblées. Elle consiste, pour un couple $(S, \mathcal{F})$, à comparer "localement" les solutions alternatives deux à deux avec la méthode majoritaire. Le vainqueur de Condorcet du couple de solutions courant est comparé à la solution suivante et ce itérativement.

Même lorsqu'il existe un vainqueur de Condorcet pour $(S, \mathcal{F})$, cette technique peut conduire à son élimination. Pis, si le président de l'assemblée maîtrise l'ordre dans lequel les solutions seront soumises au vote majoritaire, il peut parvenir à faire voter n'importe quoi (voir RAYNAUD [2000, § 2.3.2], MARCHANT *et al*. [2000]). L'ensemble de choix n'est pas conservé lorsqu'une règle d'agrégation est utilisée dans un processus itératif. Il existe bien d'autres exemples dans le domaine des procédures de vote.

Pour y remédier, il est nécessaire de considérer la hiérarchie agrégative et le processus de décision comme un tout. Ils nécessitent un contrôle au niveau global, pouvant se faire par l'analyse axiomatique. Nous renvoyons aux travaux de Marc PIRLOT [111].

4.2.2.4. Analyse Axiomatique et Règles de Choix Collectif

Des investigations axiomatiques ont aussi été menées pour les règles de choix collectif. Au moins deux voies ont été suivies :

– L'étude les propriétés intrinsèques aux règles de choix collectif (voir par exemple SEN [1986 § 5], DENICOLÓ [1999]) ;

[111] PIRLOT Marc [1994] – Why Trying to Characterize the Procedures Used for Multi-Criteria Decision Aid – Cahiers du CERO n°36 : 283-292.

– L'étude des propriétés du couple $(S, \mathcal{F})$ pour que deux règles de choix collectif RCC_1 et RCC_2 conduisent au même ensemble de choix collectif, ou aient une intersection non vide, ou admettent l'inclusion dans un sens ou dans un autre, etc. ; La comparaison d'ensembles de choix collectif a été traité par PERNY & SPANJAARD [2002] et LAHIRI [2002] dans un cadre général.

Les travaux sur le sujet sont moins nombreux que dans le cadre agrégatif. Ce sont généralement des sujets connexes qui sont traités, comme les fonctions de choix collectif (SEN [1986]) et les règles d'agrégation.

❖ Axiomes de Choix Collectif

En TCS, des travaux sporadiques ont porté sur la **caractérisation axiomatique de l'ensemble de choix collectif à partir d'un profil de relations binaires de préférences individuelles**. Il se rapproche des travaux de la section 3.4, dans un cadre collectif. Nous présentons ci-dessous quelques axiomes pour la règle de choix collectif :

Axiome de choix unanime optimal : Une règle de choix collectif *RCC* satisfait l'axiome de choix unanime si, pour tout couple $(S, \mathcal{F})$ de son domaine de définition, avec $\mathcal{F}$ un profil de relations binaires de préférence sur S, l'implication suivante est satisfaite :

[Toute solution $x \in S$ vérifiant : pour tout $y \in S$ et pour tout $i \in I_e$, $x \succcurlyeq_i y$] $\Rightarrow$ $x \in C(S, \mathcal{F})$.

Cet axiome est une manière de formuler l'absence de conflit global dans le cadre des relations binaires de préférence individuelles. C'est aussi une généralisation de l'axiome d'optimalité (formule (3–5) page 106) au cas collectif. Il en existe aussi une telle généralisation pour la maximalité : l'**axiome de choix unanime maximal**. Mais lorsque le couple $(S, \mathcal{F})$ est ne présente aucune solution ayant de telles caractéristiques, qu'est il possible de faire ? D'autres axiomes ont apparu, comme l'axiome de choix parétien (voir DENICOLÓ [1999]) :

Axiome de choix parétien (ou **axiome de Pareto optimalité**) : Une règle de choix collectif *RCC* satisfait l'axiome de choix paretien si, pour tout couple $(S, \mathcal{F})$ de son domaine de définition, avec $\mathcal{F}$ un profil de relations binaires de préférence sur S, l'implication suivante est satisfaite : [Tout couple $x, y \in S$ vérifiant : pour tout $i \in I_e$, $x \succcurlyeq_i y$ et pour au moins un $i \in I_e$, $x \succ_i y$] $\Rightarrow y \notin C(S, \mathcal{F})$.

DENICOLÓ [1999] présenta aussi des principes dédiés aux règles d'agrégation collectives, véhiculant des principes d'équité.

Nous poursuivons en nous tournant vers des axiomes de redondance lors d'un choix collectif.

❖ Redondance dans le Cadre des Ensembles de Choix Collectifs

Revenons sur les axiomes de redondance utilisés pour réduire la taille des ensembles de choix (§ 3.4.5). Les mêmes axiomes peuvent être utilisés après une agrégation du profil en une relation binaire de préférence. Mais, elles peuvent ne plus avoir la même signification au niveau global. La redondance *stricto sensu* que véhicule l'indiscernabilité au niveau de la relation binaire de préférence n'implique généralement pas la même redondance *stricto sensu* au niveau du profil. Elle peut souvent apparaître trop filtrante, approximative et radicale. La perte d'information préférentielle induite par l'agrégation en est la cause. En terme de redondance *stricto sensu*, la 'similarité pertinente' de SEN [1970] est plus attrayante (voir note de bas de page numéro 81 en page 117).

Cette branche de recherche est pour le moment en pleine expansion. Et beaucoup reste à faire pour arriver à des axiomes satisfaisants comme dans le cadre d'une seule relation binaire de préférence.

❖ Comparaison d'Ensembles de Choix Collectif

La comparaison d'ensembles de choix collectif est surtout utilisée pour la résolution : Par exemple, certaines règles sont très difficiles à utiliser tel quel, et une approximation est souvent attrayante. Une seconde utilisation se fait avec les méthodes interactives usuelles qui, pour contrôler l'exploration interactive d'ensembles de solutions, recherchent des familles de fonctions objectives agrégatives pour cibler des solutions de meilleur compromis. L'ensemble optimal des fonctions objectives est inclus dans l'ensemble de choix collectif, et cette famille d'ensembles optimaux forment un recouvrement de l'ensemble de choix. Ceci permet de se déplacer correctement dans l'ensemble de choix collectif, avec un minimum de maîtrise sur le type de solution recherché[112].

Remarquons qu'un contrôle satisfaisant peut être obtenu sans pour cela posséder de telles règles de substitution, en utilisant les problèmes de consistance (§ 7.1).

[112] Nous renvoyons à la litérature spécifique en AMCD traitant du sujet comme TEGHEM [1996] en programmation linéaire multiobjectif, MARCOTTE & SOLAND [1986], VINCKE [1989] et ROY [1985] & BOUYSSOU [1993] en programmation mathématique multiobjective. Nous en reparlerons succinctement au § 7.2.

5. Contraintes et Représentation Implicite des Préférences[113]

5.1. Introduction

5.1.1. Comment Exprimer des Préférences sur des Solutions Alternatives non Explicitement Données ?

Un problème supplémentaire se pose lorsque les solutions sont données en compréhension. Comment représenter les préférences binaires sur un ensemble de solutions donné en compréhension.

Les préférences binaires sont très bien adaptées pour simplement « comparer deux solutions données ». Pour de telles comparaisons, nous avons vu qu'il n'était pas nécessaire de faire usage d'un ensemble de valeurs totalement ordonnées comme le font les fonction-critères. D'un autre coté, l'usage des vrai-critères est pratique courante dans les problèmes comportant des solutions données en compréhension. On peut facilement généraliser cette méthodologie aux fonction-critères. Qu'en est-il des préférences binaires ?

Prenons un exemple (tiré de LANG [2002]) : Imaginons qu'un groupe de personnes doit se mettre d'accord sur un menu commun, un menu étant composé d'une entrée (5 valeurs possibles), d'un plat de résistance (6 valeurs possibles), d'un éventuel plateau de fromages (2 valeurs possibles : oui / non), d'un dessert (6 valeurs possibles) et de deux vins parmi 6 possibles (C_6^2 = 15 valeurs possibles). L'espace des candidats est ici l'ensemble S des menus possibles, qui est de cardinalité $5 \times 6 \times 2 \times 6 \times 15 = 5400 = \mathrm{card}(S)$.
Le nombre de solutions possibles étant exponentiel en le nombre de variables, il n'est pas raisonnable de demander aux votants d'ordonner ou d'évaluer explicitement chaque couple de solutions potentielles. La construction directe d'une relation binaire de préférence, où tout autre modèle basé sur une structure de préférence sur S est inconcevable.

En TCS, il est fait abstraction de ce problème. Les relations binaires de préférence sont directement fournies. En AMCD (programmation mathématique multiobjective), les vrai-critères servent d'intermédiaires indispensables pour exprimer des relations binaires de préférence.

Quelles sont les différentes manières existantes de représenter des préférences sur un ensemble de solutions donné en compréhension ?

De la même façon qu'il existe différentes façons de représenter une relation binaire en machine (matrice d'incidence, liste d'adjacence, etc.[114]) lorsque les solutions sont explicites, de même, il existe différentes façons de les coder dans un environnement implicite.

5.1.2. Modélisation de Préférences par l'Intermédiaire de Contraintes Souples

L'un des grands chantiers des problèmes combinatoires d'aide à la décision est la représentation et la récolte d'information préférentielle. Proposer des modèles qui soient les mieux

[113] Les éléments présentés dans ce chapitre ont fait l'objet de deux publications JOSEPH *et al*. [2003*a*], [2003*b*].
[114] Voir par exemple SAKAROVITCH [1984*a*], GONDRAN & MINOUX [1996], parmi d'autres.

adaptés à ce qu'un acteur veut formuler, exprimer. La quantité d'information à communiquer au système devient très vite prohibitive lorsque les solutions sont données en compréhension, étant donné que toutes les éventualités doivent être prévues. Les problèmes périodiques, répétitifs, comme la planification d'horaires de travail, ont un certain nombre d'avantages comme l'apprentissage, ou encore, l'existence d'un ensemble de contraintes et de préférences périodiques, pouvant servir de base aux acteurs pour exprimer leurs contraintes et préférences individuelles ou collectives du moment. Il n'est pas nécessaire de redécouvrir la roue à chaque fois.

Il existe actuellement au moins 3 types de modèles en PPC permettant de représenter des préférences dans un cadre implicite (§ 2.1.4.5) :

- les contraintes souples de propriété,
- les VC-contraintes souples simples dédiées aux fonctions objectives,
- les contraintes récursives (§ 2.1.4.3) sur les contraintes souples précédentes, comme les préférences conditionnelles (BOUTILIER *et al*. [1997], BOUTILIER *et al*. [1999]).

Ces 3 modélisations sont complémentaires et non exclusives. Elles sont assez limitées pour représenter des relations binaires de préférence, ainsi que les règles d'agrégations : Les contraintes souples de propriété sont pour l'instant utilisées comme attributs, et sont presqu'exclusivement agrégés via une VC-contrainte souple ; Les VC-contraintes souples ont été conçues pour représenter des vrai-critères, et leur adaptation aux relations binaires de préférence n'a pour l'instant pas été envisagée ; Les contraintes récursives (préférences conditionnelles) offrent de grandes perspectives de représentation des relations binaires de préférence. Mais pour l'instant, les applications se restreignent à de petits exemples.

5.1.3. Les Grandes Lignes du Chapitre

Nous avons décidé d'utiliser une nouvelle représentation, complémentaire, dédiée aux préférences binaires dont l'ensemble des attitudes fondamentales est donné en extension. Le reste du chapitre est donc organisé comme suit. La section 5.2 introduit trois nouvelles contraintes souples :

- la PB-contrainte souple simple (§ 5.2.1) dédiée aux relations binaires de préférence sur un ensemble de solutions donné en compréhension par un produit cartésien,
- la PB-contrainte souple étendue (§ 5.2.2) qui est une version étendue de la PB-contrainte souple simple, et
- la VC-contrainte souple étendue (§ 5.2.3) dédiée aux vrai-critères étendus sur un ensemble de solutions donné par un produit cartésien.

Nous cloturons cette section avec la description de quelques propriétés de ces contraintes souples (§5.2.4).

La section 5.3 est consacrée à l'utilisation de ces contraintes souples dans la définition de structures complexes. Tout d'abord, la sous-section 5.3.1 s'attarde sur l'utilisation de ces contraintes souples pour définir des règles d'agrégation. Ensuite, les systèmes de contraintes basés sur les préférences (P-CS) sont introduites au § 5.3.2 comme modèles de systèmes réels complexes implicites usant de relations binaires de préférence pour définir leur modèle préférentiel. Enfin, le reste de la section est consacré à la définition de problèmes d'aide à la décision à partir des P-CS : les problèmes de choix (§ 5.3.3) des solutions de meilleure qualité, les problèmes d'aide face à l'incohérence dans un contexte de choix (§ 5.3.4), et les problèmes de consistance dans un contexte de choix (§ 5.3.5).

5.2. Contraintes Souples dédiées aux Préférences Binaires à Attitude Fondamentales en Extension

Nous présentons ici un nouveau type de contraintes souples dédié aux préférences binaires (§ 3.3.1 à 3.3.3) dont les attitudes sont données en extension, comme les relations binaires de

préférence (§ 3.3.1), les relations binaires utilisant la notion de préférence faible (voir ROY [1985], ROY & BOUYSSOU [1993]), ou encore les relations binaires utilisant la logique tétravalente (§ 3.3.2.3). Ces contraintes souples ne forment pas un autre concept concurrent, mais plutôt une représentation complémentaire aux modèles existants.

5.2.1. Contraintes Souples Dédiées aux Préférences Binaires Simples

Nous traitons le cas de la relation binaire de préférence simple, mais cette contrainte souple est valable pour toute préférence binaire simple dont les attitudes sont données en extension. Nous noterons l'ensemble des attitudes d'une relation binaire de préférence *AF* au lieu de *AF*(*RBP*) comme au § 3.3.2.1, pour simplifier l'écriture. Rappelons que *AF*(*RBP*) = {mieux, indifférent, incomparable, moins bien} et est aussi noté $\{\simeq, \succ, \|, \prec\}$.

Montrons comment nous avons décidé de modéliser une relation binaire de préférence, ainsi que quelques propriétés et relations avec les autres contraintes souples.

5.2.1.1. Les PB-Contraintes Souples Simples

Étant donnés un ensemble V de variables et un ensemble D de domaines associé à V, alors une relation binaire de préférence $\succcurlyeq$ sur $\mathcal{D}_V$ peut être décrite par une contrainte $c_{\succcurlyeq}[\alpha, x]$ paramétrée par une solution x et une attitude α. L'ensemble $\{c_{\succcurlyeq}[\alpha, x], \forall\ (\alpha, x) \in \mathcal{PR}(AF) \times \mathcal{D}_V\}$ est appelé **PB-contrainte souple simple** associée à la relation binaire de préférence $\succcurlyeq$. Cette contrainte souple est aussi notée $\{c_{\succcurlyeq}[\alpha, x]\}_{\alpha, x}$. L'ensemble réalisable de $c_{\succcurlyeq}[\alpha, x]$ est noté :

$$S(c_{\succcurlyeq}[\alpha, x], V, D) = \{y \in \mathcal{D}_V \text{ tel que : } y\ \alpha_{\succcurlyeq}\ x\} \tag{5–20}$$

avec $\alpha_{\succcurlyeq}$ indiquant l'attitude α de la relation binaire de préférence $\succcurlyeq$. Par analogie aux graphes orientés, la sémantique de $c_{\succcurlyeq}[\alpha, x]$ décrit le voisinage de x dans l'ensemble $\mathcal{D}_V$ selon la relation binaire $\alpha_{\succcurlyeq}$. Ainsi, la sémantique d'une PB-contrainte souple simple consiste à répondre aux questions de voisinage associées aux attitudes de la préférence binaire modélisée. Dans le cas de la rbp ce problème est le suivant :

Voisinage : Étant donné (V, D, $\succcurlyeq$, α, x) comme défini ci-dessus, retourner une solution de $\mathcal{D}_V$ jugée $\alpha_{\succcurlyeq}$ par rapport à la solution x, si une telle solution existe. Sinon retourner "non".

Le problème de comparaison (§ 3.2.1.2) correspond alors au problème d'appartenance associé au problème Voisinage. Cette modélisation des relations binaires de préférence dans un contexte implicite est assez naturelle, comme nous le verrons avec les règles d'agrégation au § 5.3.1. De plus, elle offre de larges perspectives de résolution de problèmes (chapitre 7).

Contrairement à l'ensemble des variables V portant sur une PB-contrainte souple simple, son ensemble de solutions réalisables est paramétré par une attitude et une solution alternative.

Les PB-contraintes souples simples ont déjà été employées dans la littérature par GAVANELLI [2002*a*] et [2002*b*] dans le cas particulier de α = {mieux, indifférent, incomparable}. Cette contrainte souple permet de concevoir et d'utiliser des algorithmes plus variés et plus efficaces (voir section 7.1). Notons aussi les travaux de KOSTREVA & OGRYCZAK [1999] en programmation linéaire et OGRYCZAK [2000] en programmation linéaire en nombres entiers. Nous y reviendrons au § 7.2.

5.2.1.2. Des VC-Contraintes Souples aux PB-Contraintes Souples Simples

Une PB-contrainte souple simple $\{c_f[\alpha, x]\}_{\alpha, x}$ peut être construite à partir de n'importe quelle VC-contrainte souple c_f. Étant donné $V(c_f) = V(f) \cup \{v_f\}$, en utilisant la notation de la section 2.1.4.5, alors $V(\{c_{\succcurlyeq}[\alpha, x]\}_{\alpha, x}) = V(f)$ et sa sémantique opérationnelle est la suivante :

$$S(c_f[\alpha, x], V, D) = \{y \in \mathcal{D}_{V(f)} \text{ tel que : } f(y)\ \alpha\ f(x)\} \quad \forall\ (\alpha, x) \in \mathcal{PR}(AF) \times \mathcal{D}_{V(f)}\ . \qquad (5\text{–}21)$$

Si nous développons la PB-contrainte souple simple selon les différentes attitudes fondamentales, nous obtenons[115] :

$c_f[\simeq, x] \Leftrightarrow \{ c_f, \{D(v_f) = \{f(x)\}\} \}$	$c_f[\succ, x] \Leftrightarrow \{ c_f, \{D(v_f) \cap]f(x), \max(D(v_f))]\} \}$
$c_f[\|\|, x] = \varnothing$	$c_f[\prec, x] \Leftrightarrow \{ c_f, \{D(v_f) \cap [\min(D(v_f)), f(x)[\} \}$

Ces quatre relations définissent une transformation d'une VC-contrainte souple simple en une PB-contrainte souple simple.

Comparées à la contrainte souple de propriété, la PB-contrainte souple simple est un ensemble de contraintes de propriétés liées entre elles par un ensemble de propriétés qui sont donné au § 5.2.4.

5.2.2. Contraintes Souples Dédiées aux Préférences Binaires Étendues

Dans le cadre étendu, nous traitons aussi le cas de la relation binaire de préférence, mais cette contrainte souple est valable pour toute préférence binaire étendue dont les attitudes sont données en extension.

5.2.2.1. Les PB-Contraintes Souples Étendues

Considérons un ensemble V de variables, un ensemble D de domaines associé à V et un ensemble I_e d'individus observés (concernés). Une contrainte souple étendue sur $\mathcal{D}_V \times I_e$ peut être construite à partir de divers problèmes. Par exemple,

(a) Étant donné un quadruplet ($\succcurlyeq$, x, i, α) formé d'une relation binaire étendue $\succcurlyeq$ sur $\mathcal{D}_V \times I_e$, un élément x de $\mathcal{D}_V$, un individu observé i de I_e et une attitude α de $\mathcal{PR}(AF)$, retourner une position (y, j) de $\mathcal{D}_V \times I_e$ vérifiant $(y, j)\ \alpha_{\succcurlyeq}\ (x, i)$, si une telle position existe. Sinon retourner "non".

Ou encore,

(b) Étant donné un quadruplet ($\succcurlyeq$, x, i, α) formé d'une relation binaire étendue $\succcurlyeq$ sur $\mathcal{D}_V \times I_e$, un élément x de $\mathcal{D}_V$, un individu observé i de I_e et une attitude α de $\mathcal{PR}(AF)$, retourner un élément y de $\mathcal{D}_V$ vérifiant $(y, i)\ \alpha_{\succcurlyeq}\ (x, i)$, si un tel élément existe. Sinon retourner "non".

(c) Étant donné un quadruplet ($\succcurlyeq$, x, i, α) formé d'une relation binaire étendue $\succcurlyeq$ sur $\mathcal{D}_V \times I_e$, un élément x de $\mathcal{D}_V$, un individu observé i de I_e et une attitude α de $\mathcal{PR}(AF)$, retourner un élément y de $\mathcal{D}_V$ pour lequel il existe un concerné j avec $(y, j)\ \alpha_{\succcurlyeq}\ (x, i)$, si un tel élément existe. Sinon retourner "non".

Nous pouvons continuer à en énumérer, mais nous nous arrêtons là. Certains problèmes sont plus pertinents que d'autres selon, l'information de départ, l'utilisation qui est faite des contraintes souples, les règles d'agrégation utilisées, etc.. Ainsi, le problème (a) décrit la sémantique d'une contrainte souple notée $\{c_{\succcurlyeq}[\alpha, (x, i)]\}_{(\alpha, (x, i))}$ dont la formulation est :

$$S(c_{\succcurlyeq}[\alpha, (x, i)], V \cup \{v(I_e)\}, D \cup \{I_e\}) = \{(y, j) \in \mathcal{D}_V \times I_e, \text{ tel que : } (y, j)\ \alpha_{\succcurlyeq}\ (x, i)\} \quad (5\text{–}22)$$

avec $v(I_e)$ une nouvelle variable à laquelle on associe le domaine $D(v(I_e)) = I_e$, et $\alpha_{\succcurlyeq}$ l'attitude α de la relation binaire de préférence étendue $\succcurlyeq$. En d'autres termes, la contrainte souple $\{c_{\succcurlyeq}[\alpha, (x, i)]\}_{\alpha, (x, i)}$ est une **PB-contrainte souple simple sur l'ensemble des positions**. Elle est très bien adaptée aux règles d'agrégation simples sur l'ensemble des solutions alternatives $\mathcal{D}_V \times I_e$. Est-il pertinent de les utiliser pour modéliser d'autres règles d'agrégation ?

Étudions maintenant le cas du problème (b). Il capture uniquement l'information préférentielle de $\succcurlyeq$ propre à chaque individu observé i. L'information préférentielle entre deux posi-

[115] Pour alléger l'écriture des PB-contraintes moles, par convention, si une attitude est formée d'une seule attitude fondamentale, nous supprimons les accolades. Ainsi $c_{\succcurlyeq}[\succ, x] = c_{\succcurlyeq}[\text{mieux}, x]$ est équivalent à $c_{\succcurlyeq}[\{\text{mieux}\}, x]$.

tions concernant deux individus différents n'est pas prise en compte. Par conséquent, le problème (b) décrit la sémantique de card(I_e) PB-contraintes souples simples sur l'ensemble $\mathcal{D}_V$: $\{c_{(\succcurlyeq,\, i)}[\alpha,\, z]\}_{\alpha,\, z}$ pour tout $i \in I_e$. Si les comparaisons interpersonnelles ne sont pas utilisées dans notre problème, alors, cette représentation est suffisante.

Pour user de règles d'agrégation étendues, nous avons adopté une modélisation différente : Étant donné V un ensemble de variables et D un ensemble courant de domaines associé à V, une relation binaire de préférence étendue $\succcurlyeq$ sur le produit cartésien $\mathcal{D}_V$ de domaines de D et sur I_e est décrite par une contrainte $c_{\succcurlyeq}[\alpha, x, i, j]$ paramétrée par une attitude α, une solution x et deux individus (i, j). L'ensemble $\{c_{\succcurlyeq}[\alpha, x, i, j],\ \forall\ (\alpha, x, i, j) \in \mathcal{PR}(AF) \times \mathcal{D}_V \times I_e \times I_e\}$, noté plus succinctement $\{c_{\succcurlyeq}[\alpha, x, i, j]\}_{(\alpha,\, x,\, i,\, j)}$ est appelé **PB-contrainte souple étendue**. En notant $\alpha_{\succcurlyeq}$ l'attitude α de la relation binaire de préférence étendue $\succcurlyeq$, l'ensemble des solutions réalisables de chaque contrainte souple $c_{\succcurlyeq}[\alpha, x, i, j]$ est :

$$S(c_{\succcurlyeq}[\alpha, x, i, j],\ V, D) = \{y \in \mathcal{D}_V \text{ tel que : } (y, j)\ \alpha_{\succcurlyeq}\ (x, i)\}\ . \qquad (5\text{–}23)$$

Cette définition correspond exactement au problème (c) présenté ci-dessus.

5.2.2.2. PB-Contraintes Souples Étendues et Autres Contraintes Souples

Quelle est la relation entre $\{c_{\succcurlyeq}[\alpha, x, i, j]\}_{(\alpha,\, x,\, i,\, j)}$ et $\{c_{\succcurlyeq}[\alpha, (x, i)]\}_{(\alpha,\, (x,\, i))}$? Les ensembles $S(c_{\succcurlyeq}[\alpha, x, i, j],\ V, D) \times \{j\}$ pour tout $j \in I_e$ forment une partition de $S(c_{\succcurlyeq}[\alpha, (x, i)],\ V \cup \{v(I_e)\}, D \cup \{I_e\})$:

$$S(c_{\succcurlyeq}[\alpha, (x, i)],\ V \cup \{v(I_e)\}, D \cup \{I_e\}) = \bigcup_{j \in I_e} S(\, c_{\succcurlyeq}[\alpha, x, i, j],\ D) \times \{j\}\ . \qquad (5\text{–}24)$$

$$S(c_{\succcurlyeq}[\alpha, x, i, j], D) = \{y \in \mathcal{D}_V \text{ tel que : } (y, j) \in S(c_{\succcurlyeq}[\alpha, (x, i)],\ V \cup \{v(I_e)\}, D \cup \{I_e\})\}. \qquad (5\text{–}25)$$

La PB-contrainte souple étendue $\{c_{\succcurlyeq}[\alpha, x, i, j]\}_{(\alpha,\, x,\, i,\, j)}$ est plus précise que la contrainte souple simple $\{c_{\succcurlyeq}[\alpha, (x, i)]\}_{(\alpha,\, (x,\, i))}$ sur les positions. Mais selon la rbp étendue modélisée et l'utilisation qui en est faite, une représentation sera plus pertinente ou/et efficace que l'autre. Nous verrons aux § 7.4.4 et § 7.4.5 quelques exemples d'utilisation.

Comment se traduit la relation entre les préférences individuelles simples et leurs homologues étendues[116] à travers la PB-contrainte souple simple et sa version étendue ? En notant $\{c_i[\alpha, x]\}_{(\alpha,\, x)}$ et $\{c_i[\alpha, x, i, j]\}_{(\alpha,\, x,\, i,\, j)}$ respectivement la PB-contrainte souple simple et la version étendue de l'individu i, alors :

$$c_i[\alpha, x, i, i] = c_i[\alpha, x] \ \text{ pour tout } (\alpha, x, i) \in \mathcal{PR}(AF) \times \mathcal{D}_V \times I_e\ . \qquad (5\text{–}26)$$

Par conséquent, sachant qu'à toute règle d'agrégation simple correspond une règle d'agrégation étendue, il suffit de remplacer toute occurrence de la contrainte $\{c_i[\alpha, x]\}_{(\alpha,\, x)}$ par la contrainte $\{c_i[\alpha, x, i, i]\}_{(\alpha,\, x,\, i,\, i)}$ pour obtenir exactement le même résultat, mais en utilisant des préférences étendues, comme nous verrons au § 7.4.4 avec la règle de Pareto.

Pour clôturer cette sous-section, notons qu'il existe d'autres possibilités de construction des préférences de type sympathie étendue. Celles-ci restent à découvrir, et sont plus adaptés que d'autres aux problèmes étudiés. Nous nous cantonnons ici aux PB-contraintes souples étendues.

Remarque 1 : (extensions) Bien que dédiées aux préférences binaires dont les attitudes fondamentales sont données en extension, rien n'empêche de concevoir des PB-contraintes souples pour d'autres relations binaires déduites directement des préférences binaires étudiées, comme l'indiscernabilité parfaite (§ 3.4.5) associée à une rbp $\succcurlyeq$ donnée. Les PB-

[116] Voir formules (4-15) page 129 ou la formule plus générale (4–18) page 135.

contraintes souples peuvent être utilisées aussi lorsque les attitudes fondamentales sont données en compréhension, en utilisant des intervalles d'attitudes par exemple.

5.2.3. *Les VC-Contraintes Souples Étendues*

Nous n'avons pas abordé le cas des vrai-critères étendus f sur $\mathcal{D}_{V(f)} \times I_e$. Comment peut-on définir une contrainte souple pour les vrai-critères étendus ? Il existe aussi plusieurs possibilités de modélisation si la même méthodologie que pour les vrai-critères simples est utilisée. Par exemple, une unique variable valeur peut être associée à f, ou à chaque individu observé. Nous avons décidé de définir une variable valeur $v_f(i)$ par individu i de I_e, prenant ses valeurs dans l'ensemble des réels[117] ($\Leftrightarrow D(v_f(i)) \subseteq \quad , \forall\ i$). Le vrai-critère étendu f est alors modélisé par l'ensemble des contraintes $\{c_f[i], \forall\ i \in I_e\}$ appelé **VC-contrainte souple étendue** et noté plus simplement $\{c_f[i]\}_i$. Chaque contrainte $c_f[i]$ porte sur les variables $V(f) \times \{v_f(i)\}$, et vérifie : $S(c_f[i], D) = \{(y_1, y_2) \in \mathcal{D}_{V(f)} \times D(v_f(i))$ tel que : $f(y_1, i) = y_2\}$.

Peut-on créer une PB-contrainte souple étendue à partir d'une VC-contrainte souple étendue ? Une PB-contrainte souple étendue $\{c_f[\alpha, x, i, j]\}_{(\alpha, x, i, j)}$ peut être construite à partir de n'importe quelle VC-contrainte souple étendue $\{c_f[i]\}_i$ portant sur les variables $V(f) \cup \{v_f\}$. Son ensemble de variables est alors $V(\{c_f[\alpha, x, i, j]\}_{(\alpha, x, i, j)}) = V(f)$ et sa sémantique est la suivante :

$$S(c_f[\alpha, x, i, j], V, D) = \{y \in \mathcal{D}_{V(f)} \text{ tel que : } f(y, i)\ \alpha\ f(x, j)\} \qquad (5\text{–}27)$$
$$\forall\ (\alpha, x, i, j) \in \mathcal{PR}(AF) \times \mathcal{D}_{V(f)} \times I_e^2\ .$$

5.2.4. *Quelques Propriétés des PB-Contraintes Souples*

Les PB-contraintes souples simples $\{c_k[\alpha, y]\}_{\alpha, y}$ et étendues $\{c_k[\alpha, y, i, j]\}_{(\alpha, y, i, j)}$ sont composées de contraintes souples liées par des propriétés entre leur sémantique. Ces propriétés sont pertinentes, notamment pour réduire le nombre d'algorithmes de filtrage à concevoir dans un but opérationnel (comme nous verrons au § 7.3), ou encore pour résoudre des problèmes d'aide à la décision par leur intermédiaire (§ 7.1 et 7.4). Deux types de propriétés sont étudiés ici : Celles liées aux attitudes et celles liées aux propriétés structurelles des préférences binaires étudiées.

5.2.4.1. *Propriétés des Attitudes et Solutions Réalisables des Contraintes Souples*

Les relations entre les différentes attitudes peuvent être exploitées pour établir des relations entre les ensembles de solutions réalisables des contraintes souples constituant les PB-contraintes souples.

Par définition, toute attitude est formée d'attitudes fondamentales. Par conséquent, pour toute solution fixée, la contrainte souple associée à une attitude peut, de la même manière, être décomposée. Les propriétés suivantes expriment que les opérations ensemblistes sur $\mathcal{P}(AF)$ sont compatibles avec les opérations ensemblistes sur $\{S(c_k[\alpha, x], V, D)$ tel que : $\alpha \subseteq AF\}$.

Propriétés 2 : Étant donnée une PB-contrainte souple simple $\{c_k[\alpha, y]\}_{\alpha, y}$ émise par un individu k sur un ensemble de variables V associé à l'ensemble de domaines D, et quelque soit $x \in \mathcal{D}_V$, alors :

En notant par convention, $S(c_k[AF, x], V, D) = \mathcal{D}_V$, et $S(c_k[\varnothing, x], V, D) = \varnothing$, alors pour tout $\alpha_1, \alpha_2 \in \mathcal{P}(AF)$, nous avons :

[117] La remarque donnée en note en bas de page numéro 17 en page 43 est aussi valable ici.

(a) $S(c_k[\alpha_1 \cup \alpha_2, x], V, D) = S(c_k[\alpha_1, x], V, D) \cup S(c_k[\alpha_2, x], V, D)$.
(b) $S(c_k[\alpha_1 \cap \alpha_2, x], V, D) = S(c_k[\alpha_1, x], V, D) \cap S(c_k[\alpha_2, x], V, D)$.
(c) $\alpha_1 \subseteq \alpha_2 \Rightarrow S(c_k[\alpha_1, x], V, D) \subseteq S(c_k[\alpha_2, x], V, D)$.

Preuve: Montrons l'assertion (a), les autres assertions se démontrant de manière similaire. Pour alléger la notation, nous omettons de faire référence à l'individu émetteur dans les préférences. C'est-à-dire qu'au lieu d'écrire "$y\ (\alpha)_k\ x$" qui signifie "y est jugé α par rapport à x selon l'individu k", nous écrirons "$y\ \alpha\ x$". Partons de la définition de l'ensemble réalisable de $c_k[\alpha_1 \cup \alpha_2, x]$:

$$\begin{aligned} S(c_k[\alpha_1 \cup \alpha_2, x], V, D) &= \{y \in \mathcal{D}_V \text{ tel que : } y\ (\alpha_1 \cup \alpha_2)\ x\} \\ &= \{y \in \mathcal{D}_V \text{ tel que : } y\ \alpha_1\ x \text{ ou } y\ \alpha_2\ x\} \\ &= \{y \in \mathcal{D}_V \text{ tel que : } y\ \alpha_1\ x\} \cup \{y \in \mathcal{D}_V \text{ tel que : } y\ \alpha_2\ x\} \end{aligned}$$

Le second membre de cette dernière égalité permet de démontrer l'assertion. ❑

Un ensemble de propriétés équivalentes est valable pour les PB-contraintes souples étendues :

Propriétés 3 : Considérons une PB-contrainte souple étendue $\{c_k[\alpha, y, i, j]\}_{(\alpha, y, i, j)}$ émise par un individu k sur un ensemble de variables V associé à l'ensemble de domaines D, et quelque soient $x \in \mathcal{D}_V$ et $(i, j) \in I_e \times I_e$, alors :

En notant par convention, $S(c_k[AF, x, i, j], V, D) = \mathcal{D}_V$, et $S(c_k[\varnothing, x, i, j], V, D) = \varnothing$, alors pour tout $\alpha_1, \alpha_2 \in \mathcal{P}(AF)$, nous avons :

(a) $S(c_k[\alpha_1 \cup \alpha_2, x, i, j], V, D) = S(c_k[\alpha_1, x, i, j], V, D) \cup S(c_k[\alpha_2, x, i, j], V, D)$
(b) $S(c_k[\alpha_1 \cap \alpha_2, x, i, j], V, D) = S(c_k[\alpha_1, x, i, j], V, D) \cap S(c_k[\alpha_2, x, i, j], V, D)$
(c) $\alpha_1 \subseteq \alpha_2 \Rightarrow S(c_k[\alpha_1, x, i, j], V, D) \subseteq S(c_k[\alpha_2, x, i, j], V, D)$.

Preuve: La démonstration des différentes assertions est équivalente à celles des Propriétés 2. ❑

En définitive, ces propriétés informent que l'ensemble $\{S(c_k[\alpha, x], V, D)\ \ \forall\ \alpha \subseteq AF\}$ (resp. l'ensemble $\{S(c_k[\alpha, x, i, j], V, D)\ \ \forall\ \alpha \subseteq AF\}$ muni de l'inclusion est structuré en treillis[118].

5.2.4.2. Propriétés Structurelles et Solutions Réalisables des Contraintes Souples

Nous nous limitons ici aux PB-contraintes souples simples représentant une relation binaire de préférence. Lorsque le paramètre x n'est pas fixé, les ensembles de solutions réalisables des contraintes souples constituant les PB-contraintes souples simples possèdent des propriétés dépendantes de la structure de la relation binaire de préférence modélisée. Ainsi, nous présentons trois ensembles de propriétés : les propriétés engendrées par (*a*) la transitivité, (*b*) l'acyclicité et (*c*) l'optimalité et la maximalité :

Propriétés 4 (transitivité et PB-contrainte souple simple) : Étant donnée une relation binaire de préférence $\succcurlyeq$,

(a) Si la relation d'indifférence $\simeq$ est transitive, alors :
$$x \simeq y \Rightarrow S(c_{\succcurlyeq}[\text{indifférent}, x], V, D) = S(c_{\succcurlyeq}[\text{indifférent}, y], V, D).$$
(b) Si la relation d'incomparabilité $\|$ est transitive, alors :
$$x \parallel y \Rightarrow S(c_{\succcurlyeq}[\text{incomparable}, x], V, D) = S(c_{\succcurlyeq}[\text{incomparable}, y], V, D).$$
(c) Si la relation de préférence stricte $\succ$ est transitive, alors :
$$x \succ y \Rightarrow S(c_{\succcurlyeq}[\text{mieux}, x], V, D) \subset S(c_{\succcurlyeq}[\text{mieux}, y], V, D).$$
L'inclusion est stricte car $x \in S(c_{\succcurlyeq}[\text{mieux}, y], V, D)$, alors que $x \notin S(c_{\succcurlyeq}[\text{mieux}, x], V, D)$.
De plus, pour l'aversion stricte : $x \prec y \Rightarrow S(c_{\succcurlyeq}[\text{pire}, x], V, D) \subset S(c_{\succcurlyeq}[\text{pire}, y], V, D)$.

[118] Pour un approfondissement du concept de treillis voir TROTTER [1992].

(d) Si la relation de surclassement $\succcurlyeq$ est transitive, alors des propriétés (a) à (c) ainsi que de la relation (3–5) en page 106 nous déduisons que :

$$x \succcurlyeq y \Rightarrow S(c_{\succcurlyeq}[\text{surclasse}, x], V, D) \subseteq S(c_{\succcurlyeq}[\text{surclasse}, y], V, D).$$

Propriétés 5 (acyclicité et PB-contrainte souple simple) : Étant donnée une relation binaire de préférence $\succcurlyeq$,

(a) Si $\succcurlyeq$ est strictement P-acyclique alors :

$$x \succ y \Rightarrow S(c_{\succcurlyeq}[\text{mieux}, y], V, D) \subseteq S(c_{\succcurlyeq}[\{\text{surclasse, incomparable}\}, x], V, D).$$

(b) Si $\succcurlyeq$ est P-acyclique alors :

$$x \succ y \Rightarrow S(c_{\succcurlyeq}[\text{mieux}, y], V, D) \subseteq S(c_{\succcurlyeq}[\text{surclasse}, x], V, D).$$

Propriétés 6 (optimalité, maximalité et PB-contrainte souple simple) : Quelque soit les propriétés structurelles de la relation binaire de préférence $\succcurlyeq$, alors :

(a) x est une solution optimale $\Leftrightarrow \begin{cases} \forall\, y \in \mathcal{D}_V, x \in S(c_{\succcurlyeq}[\text{surclasse}, y], V, D) \Leftrightarrow \\ S(c_{\succcurlyeq}[\text{sousclasse}, x], V, D) = \mathcal{D}_V \Leftrightarrow \\ S(c_{\succcurlyeq}[\{\text{mieux, incomparable}\}, x], V, D) = \varnothing \end{cases}$

(b) Si x est une solution optimale alors $B(\mathcal{D}_V, \succcurlyeq) \subseteq S(c_{\succcurlyeq}[\text{indifférent}, x], V, D)$. Si $\succcurlyeq$ est transitive alors il y a égalité.

(c) x est une solution maximale $\Leftrightarrow \begin{cases} \forall\, y \in \mathcal{D}_V, \\ \quad x \in S(c_{\succcurlyeq}[\{\text{surclasse, incomparable}\}, y], V, D) \Leftrightarrow \\ S(c_{\succcurlyeq}[\{\text{sousclasse, incomparable}\}, x], V, D) = \mathcal{D}_V \Leftrightarrow \\ S(c_{\succcurlyeq}[\text{mieux}, x], V, D) = \varnothing \end{cases}$

(d) Si x est une solution maximale alors $M(\mathcal{D}_V, \succcurlyeq) \subseteq S(c_{\succcurlyeq}[\{\text{indifférent, incomparable}\}, x], V, D)$.

Toutes ces propriétés découlent de la définition de la transitivité, de l'acyclicité, de l'optimalité et de la maximalité, ainsi que des propositions données en début de paragraphe. Par conséquent, les démonstrations ne sont pas présentées ici[119].

Ces propriétés forment les bases de validation des algorithmes de résolution de problèmes d'aide au choix dont les instances admettent comme modèle de préférence une PB-contrainte souple simple, comme nous le verrons au § 7.1.

5.3. Systèmes de Contraintes Basés sur les Préférences et Problèmes d'Aide à la Décision

Dans cette section, les contraintes souples sont utilisées pour concevoir des instances de problèmes de choix basés sur les modèles hiérarchiques de préférence, et les systèmes de contraintes. Nous présentons aussi succinctement quelques problématiques d'aide à la décision associées.

5.3.1. Contraintes Souples pour Règles d'Agrégation et Axiomes

5.3.1.1. Des Règles d'Agrégation aux Contraintes Souples

Une règle d'agrégation est une relation entre les préférences individuelles et les préférences collectives. Les contraintes modélisant les préférences collectives sont appelées **contraintes souples collectives**, et celles modélisant les préférences individuelles sont appelées **contraintes souples individuelles**. Les VC-contraintes souples et les PB-contraintes souples sont bien adaptées à la modélisation des règles d'agrégation dans un contexte implicite. Ainsi :

[119] Pour des résultats similaires, voir SUZUMURA [1983, théorème 2.10]

Les VC-contraintes souples permettent de construire des modèles de préférence basés sur les fonctions objectives (KEENEY & RAÏFFA [1976], KEENEY [1988]) en représentant chaque règle d'agrégation par une VC-contrainte souple. La règle d'agrégation f sur un profil de fonctions objectives individuelles $\mathcal{F} = \{f_1, \ldots, f_n\}$ est traduite en VC-contrainte souple c_f comme suit :

$$V(c_f) = \{v(f_1), \ldots, v(f_n)\} \cup \{v_f\}$$
$$S(c_f) = \{(x_1, x_2) \in \mathcal{D}_{V(\mathcal{F})} \times D(v_f) \text{ tel que } f(x_1) = x_2\}$$

avec $\{v(f_1), \ldots, v(f_n)\} = V(\mathcal{F})$ l'ensemble des variables valeurs associé à $\mathcal{F}$, v_f la variable valeur collective et $\mathcal{D}_{V(\mathcal{F})} = D(v(f_1)) \times \ldots \times D(v(f_n))$. L'utilisation de la VC-contrainte souple simple a déjà été effectuée dans la communauté « Contraintes » (par exemple RÉGIN & RUEHER [2000], RÉGIN *et al*. [2001]).

La PB-contrainte souple simple peut aussi être employer comme un composant de modèles de préférence admettant des préférences binaires et des fonctions objectives simples comme préférences individuelles. Les préférences individuelles peuvent être simples et sont notées $\{c_i[\alpha, z]\}_{\alpha, z}$ pour tout individu $i \in I_e = \{1, \ldots, n\}$, ou étendues et sont alors notées $\{c_k[\alpha, z, i, j]\}_{(\alpha, z, i, j)}$ pour tout individu $k \in I_{se} = \{1, \ldots, n\}$. Par contre les préférences collectives sont simples. A la règle d'agrégation RA est associée une PB-contrainte souple simple notée $\{c_{RA}[\alpha, z]\}_{\alpha, z}$, ou $\{c_{\succcurlyeq}[\alpha, z]\}_{\alpha, z}$ si les préférences collectives de RA sont structurées en une relation binaire de préférence simple $\succcurlyeq$. L'ensemble des variables sur lequel porte $\{c_{RA}[\alpha, z]\}_{\alpha, z}$ est égal à l'union des ensembles de variables des préférences individuelles.

$$V(\{c_{RA}[\alpha, z]\}_{\alpha, z}) = V(\{c_1[\alpha, z]\}_{\alpha, z}) \cup \ldots \cup V(\{c_n[\alpha, z]\}_{\alpha, z}) \text{ (cas simple)}$$
$$V(\{c_{RA}[\alpha, z]\}_{\alpha, z}) = V(\{c_1[\alpha, z, i, j]\}_{(\alpha, z, i, j)}) \cup \ldots \cup V(\{c_n[\alpha, z, i, j]\}_{(\alpha, z, i, j)}) \text{ (cas étendu)} .$$

Pour alléger l'écriture, notons $V = V(\{c_{RA}[\alpha, z]\}_{\alpha, z})$ l'ensemble des variables sur lequel porte $\{c_{RA}[\alpha, z]\}_{\alpha, z}$, et $\mathcal{D}_{V(RA)}$ le produit cartésien des domaines $D(v)$ pour tout $v \in V$. La sémantique de $\{c_{RA}[\alpha, z]\}_{\alpha, z}$ en (α, x) correspond aux éléments y de $\mathcal{D}_{V(RA)}$ jugés α par rapport à x selon les préférences collectives de RA :

$$S(c_{RA}[\alpha, x], V, D) = \{y \in \mathcal{D}_{V(RA)} \text{ tel que : } y\ \alpha_{RA}\ x\} \quad \forall\ (\alpha, x) \in \mathcal{PR}(AF) \times \mathcal{D}_{V(RA)}$$

avec et α_{RA} l'attitude α associée aux préférences collectives binaires de RA. L'utilisation des relations (4–16) et (4–17) en page 134 permet ensuite d'exprimer la contrainte souple collective en fonction des contraintes souples individuelles. Dans le cas d'une règle d'agrégation simple RA, tout algorithme de filtrage pour la contrainte $c_{RA}[\alpha, x]$ devra utiliser les éléments $c_i[\alpha_1, z]$, $\forall\ (\alpha_1, z) \in \mathcal{PR}(AF) \times \mathcal{D}_{V(RA)}$ pour conserver leur généralité. Dans le cas étendu, l'algorithme utilisera les contraintes souples $c_k[\alpha_1, z, i, j]$, $\forall\ (\alpha_1, z, i, j) \in \mathcal{PR}(AF) \times \mathcal{D}_{V(RA)} \times I_e^2$. Par contre, pour une meilleure efficacité algorithmique les algorithmes peuvent être spécialisés.

Comme son homologue simple, la PB-contrainte souple étendue est très bien adaptée à l'agrégation. A chaque règle d'agrégation <u>simple</u> RA est associée une PB-contrainte souple étendue notée $\{c_{RA}[\alpha, z, i, j]\}_{(\alpha, z, i, j)}$. Les préférences individuelles sont notées $\{c_k[\alpha, z, i, j]\}_{(\alpha, z, i, j)}$ pour tout individu $k \in I_{se} = \{1, \ldots, n\}$. L'ensemble des variables sur lequel porte $\{c_{RA}[\alpha, z, i, j]\}_{(\alpha, z, i, j)}$ et sa sémantique sont obtenus comme suit :

$$V(\{c_{RA}[\alpha, z, i, j]\}_{(\alpha, z, i, j)}) = V(\{c_1[\alpha, z, i, j]\}_{(\alpha, z, i, j)}) \cup \ldots \cup V(\{c_n[\alpha, z, i, j]\}_{(\alpha, z, i, j)})$$

$$S(c_{RA}[\alpha, x, i, j], V, D) = \{y \in \mathcal{D}_{V(RA)} \text{ tel que : } (y, j)\ \alpha_{RA}\ (x, i)\}$$
$$\forall\ (\alpha, x, i, j) \in \mathcal{PR}(AF) \times \mathcal{D}_{V(RA)} \times I_e^2$$

avec $\mathcal{D}_{V(RA)}$ le produit cartésien des domaines $D(v)$ pour tout $v \in V(\{c_{RA}[\alpha, z, i, j]\}_{(\alpha, z, i, j)}) = V$ et α_{RA} l'attitude α associée aux préférences collectives binaires de RA. L'utilisation de la relation (4–17) permet ensuite d'exprimer la contrainte souple collective en fonction des contraintes souples individuelles.

Le tableau suivant récapitule le type d'agrégation liant les différents types de contraintes souples individuelles aux différents types de contraintes souples collectives.

		contraintes souples collectives		
		$(v(f), c_f)$	$\{c_{RA}[\alpha, z]\}_{\alpha, z}$	$\{c_{RA}[\alpha, z, i, j]\}_{(\alpha, z, i, j)}$
contraintes souples individuelles	$(v(f_i), c_i)$	agrégation simple	agrégation simple	agrégation simple
	$\{c_i[\alpha, z]\}_{\alpha, z}$	agrégation simple	agrégation simple	agrégation simple
	$\{c_k[\alpha, z, i, j]\}_{(\alpha, z, i, j)}$	agrégation étendue	agrégation étendue	agrégation simple

Remarquons que les PB-contraintes souples restent pleinement utilisables lorsque les attitudes sont spécifiques à chaque individu d'un profil. Il en est de même lorsque les attitudes sont différentes entre les préférences individuelles et les préférences collectives.

Les PB-contraintes souples dédiées aux règles d'agrégation sont, comme l'opérateur de cardinalité (VAN HENTENRYCK & DEVILLE [1991]), des abstractions : leur définition est récursive. Les premières sont spécifiques aux modèles de préférence, alors que l'opérateur de cardinalité est dédié aux modèles de faisabilité. Par conséquent, elles augmentent énormément le pouvoir de modélisation des modèles qui les utilisent. L'utilisation des contraintes globales permettra, comme nous le verrons au chapitre 7, d'obtenir une grande efficacité de résolution. Différentes règles d'agrégation seront ainsi étudiées : la dominance de Pareto (§ 7.4.1), la règle de Pareto (§ 7.4.2 et § 7.4.4), la dominance de Lorenz (§ 7.4.3) et la règle d'équité de Suppes-Sen (§ 7.4.5).

5.3.1.2. Contraintes Souples et Axiomes

Les contraintes souples peuvent être utilisées pour exprimer directement certains axiomes d'agrégation. Par exemple, l'attitude fondamentale "mieux" de la règle de Pareto correspond à l'axiome de Pareto (§ 4.2.2.2), et se traduit directement par la contrainte $c_P[\text{mieux}, x]$, avec $S(c_P[\text{mieux}, x], V, D) = \{y \in \mathcal{D}_V \text{ tel que : } y \succ_P x\} \quad \forall\, x \in \mathcal{D}_V$. Cette possibilité élargit le champ d'applications.

Plus précisément, les axiomes qui expriment des dépendances entre les contraintes souples collectives et les contraintes souples individuelles sont modélisables par des contraintes souples. Par contre les axiomes comme IIPC (indépendance face aux comparaisons personnelles non pertinentes (§ 4.2.2.2)) ne le sont pas. L'axiome IIPC donne quelques propriétés de la sémantique de la contrainte souple collective en α et x.

De plus, toute règle d'agrégation composée d'union d'axiomes se traduit aussi par le fait que la contrainte souple associée à cette règle d'agrégation est égale à l'union des contraintes souples associées aux axiomes.

Remarquons enfin que si une règle d'agrégation $RA1$ est une extension d'une autre règle d'agrégation $RA2$ alors :

$$\begin{cases} S(c_{RA2}[\text{surclasse}, x], V, D) \subseteq S(c_{RA1}[\text{surclasse}, x], V, D) \\ S(c_{RA2}[\text{mieux}, x], V, D) \subseteq S(c_{RA1}[\text{mieux}, x], V, D) \end{cases}, \text{ pour tout } x \in \mathcal{D}_V.$$

En poursuivant dans ce sens, les différentes opérations sur les relations binaires (§ 3.3.1.1) impliquent des relations entre les ensembles réalisables qui peuvent être exploitées pour une résolution.

5.3.2. Modèles de Préférence et Systèmes de Contraintes

Ainsi, un modèle de préférence basé sur les contraintes est constitué de contraintes souples de propriété, de VC-contraintes souples et de PB-contraintes souples. Le graphe d'agrégation (§ 4.2.1.5) permet de représenter leur dépendance. La contrainte souple simple représentant la règle d'agrégation finale, appelée **contrainte souple finale**, a une sémantique impliquant toutes les autres contraintes souples constituant le modèle de préférence. C'est, en quelque sorte l'arbre qui cache la forêt. Son apparente simplicité peut cacher un modèle de préférence très complexe obtenu via une étape essentielle et difficile d'acquisition des préférences (voir les hiérarchies agrégatives au § 4.2.1.5).

En conséquence, un **système de contraintes basé sur les préférences** (*preference-based constraint system*, noté P-CS) est un couple $(\mathcal{R}, \{c_{\succcurlyeq}[\alpha, z]\}_{\alpha, z})$, où $\mathcal{R}$ est un système de contraintes décrivant l'ensemble des solutions et l'ensemble des solutions réalisables, et $\{c_{\succcurlyeq}[\alpha, z]\}_{\alpha, z}$ la contrainte souple finale décrivant les préférences des acteurs et modélisant une relation binaire de préférence $\succcurlyeq$.

5.3.3. Problèmes d'Aide au Choix et Systèmes de Contraintes Basés sur les Préférences

Plusieurs problèmes de choix peuvent être définis sur un système de contraintes basé sur les préférences. En voici un exemple :

P-CSP (*Preference-based constraint search problem* = **problème de recherche sur les contraintes basé sur les préférences**) : Étant donné un système de contraintes basé sur les préférences $(\mathcal{R}, \{c_{\succcurlyeq}[\alpha, z]\}_{\alpha, z})$, retourner :

$$\begin{cases} \text{une solution optimale pour } (S_{\mathrm{CSP}}(\mathcal{R}), \succcurlyeq) \text{ avec le label “optimal”, si } B(S_{\mathrm{CSP}}(\mathcal{R}), \succcurlyeq) \neq \varnothing, \\ \text{une solution maximale pour } (S_{\mathrm{CSP}}(\mathcal{R}), \succcurlyeq) \text{ avec le label “maximal”, si } M(S_{\mathrm{CSP}}(\mathcal{R}), \succcurlyeq) \neq \varnothing, \\ \text{une solution réalisable pour } S_{\mathrm{CSP}}(\mathcal{R}) \text{ avec le label “réalisable”, si une telle solution existe,} \\ \text{“non”, sinon.} \end{cases}$$

Ce problème recherche une solution de meilleure qualité et nous donne quelques indications sur la qualité de la solution retournée (optimalité, maximalité ou faisabilité). D'autres axiomes de choix (§ 3.4) peuvent être pris en compte pour améliorer la pertinence de l'information retournée.

Des problèmes partiels peuvent être définis à partir de P-CSP, retournant par exemple une solution maximale sinon “non”. Ensuite, ces problèmes peuvent être spécialisés (sous-problèmes) selon les propriétés de $\succcurlyeq$. De cette manière, on définit PPO-CSP (*partial preorder-based constraint search problem*) le sous-problème de P-CSP admettant uniquement les instances dont la relation binaire de préférence $\succcurlyeq$ est un préordre partiel, ou le sous-problème quasi-transitif (QT-CSP), ou encore le sous-problème acyclique (A-CSP), et ainsi de suite. Ces exemples de sous-problèmes retournent soit une solution optimale, soit une solution maximale, soit “non”, parce que l'existence d'une solution réalisable certifie au moins l'existence d'une solution maximale (voir § 3.4.1). Les problèmes usuels sur les contraintes, comme les CSP Valués (VCSP) ou les SCSP (BISTARELLI *et al*. [1999]) ou encore le *partially-ordered constraint optimization problem* de GAVANELLI ([2001], [2002*a*], [2002*b*]) sont des sous-problèmes du P-CSP. Nous n'effectuons pas la preuve ici.

La résolution d'un problème combinatoire basé sur les préférences n'est pas limitée à la recherche d'une unique solution réalisable possédant éventuellement des propriétés remarquables (voir § 3.4), si une telle solution existe. Il est nécessaire, dans le cas général, de proposer des outils algorithmiques permettant d'explorer intelligemment l'ensemble de choix. Les outils interactifs sont très bien adaptés à ce type de tâches. Dans le cas particulier où nous avons des garanties sur la taille de ces ensembles remarquables, et que le problème l'autorise, le problème d'énumération des solutions de tout un ensemble de choix peut être envisagé. Nous appelons cette version énumérative de P-CSP, le **problème de choix basé sur les contraintes et les préférences** (P-CCP = *preference-based constraint choice problem*). Comme nous

avons vu au § 3.4.2, l'énumération peut concerner seulement des sous-ensembles remarquables de l'ensemble optimal ou maximal, comme les ensembles de choix maximisants stables maximaux, ou stables parfaits. Les nouveaux problèmes obtenus sont alors identifiés selon les propriétés de l'ensemble de choix recherché. Par exemple SM-P-CCP pour le problème d'énumération d'un ensemble de choix maximisant stable maximal, basé sur les contraintes et les préférences.

Au § 7.1, nous présentons des algorithmes de résolution de certains problèmes de choix.

5.3.4. Choix et Incohérence dans les Systèmes de Contraintes Basés sur les Préférences

5.3.4.1. La Problématique d'Aide face à l'Incohérence en Situation de Choix

Ces problèmes d'aide au choix peuvent être couplés à une problématique d'aide face à l'incohérence (§ 3.2.3.5), de manière à retourner en cas d'incohérence une réponse à la fois plus explicative que le simple "non", et pertinente pour les préférences exprimées.

Partons donc de notre instance initiale ($\mathcal{R} = (V, D, C)$, $\{c_{\succcurlyeq}[\alpha, z]\}_{\alpha, z}$). Comme nous l'avons vu, lors d'inconsistance, plusieurs explications sont possibles : Le problème d'explication de l'incohérence retourne un sous-ensemble de contraintes $C' \subseteq C$ contradictoires. Le problème de relaxation cohérente retourne un ensemble de contraintes $C' \subseteq C$ tel que le problème de satisfaction de contraintes en $\mathcal{R}' = (V, D, C')$ soit consistant. Le problème de satisfaction de contraintes relaxables, retourne une solution x non réalisable de $\mathcal{R}$, sachant que $S_{\text{CSP}}(\mathcal{R}) = \varnothing$, et un sous-ensemble C' de C tel que $x \in S_{\text{CSP}}(\mathcal{R}')$ avec $\mathcal{R}' = (V, D, C \setminus C')$.

Ces problèmes ne tiennent cependant pas compte des préférences disponibles sur les couples de solutions du système de contraintes. Une possibilité est de faire usage de l'interaction homme-machine pour concevoir une nouvelle instance ($\mathcal{R}'$, $\{c_{\succcurlyeq'}[\alpha, z]\}_{\alpha, z}$) de problème de choix, à partir de l'instance initiale ($\mathcal{R}$, $\{c_{\succcurlyeq}[\alpha, z]\}_{\alpha, z}$), tel que $S_{\text{CSP}}(\mathcal{R}')$ est non vide. C'est d'ailleurs dans un tel cadre que le terme « problème sur-contraint » a été introduit. Les problèmes sur-contraints permettent de combler ce manque, en convertissant certaines contraintes dures de C en contraintes souples de propriété C_{soft} via un dialogue avec les utilisateurs, puis en agrégeant les contraintes de C_{soft} avec une VB-contrainte souple de type somme pondérée, les poids étant eux aussi obtenus par un dialogue. La présence de la relation binaire de préférence $\succcurlyeq$ nécessite en plus que C_{soft} soit agrégée avec $\succcurlyeq$ pour obtenir $\succcurlyeq'$. Cette synthèse préférentielle se fait aussi par un dialogue.

Mais, un gain considérable en temps de calcul peut être obtenu en traitant ces deux problèmes (problème de choix et recherche d'explication) simultanément, comme faisait remarquer VERFAILLIE & LOBJOIS [1999]. Cependant, comment tenir réellement compte des préférences fournies sur $\mathcal{D}_V$ (via $\{c_{\succcurlyeq}[\alpha, z]\}_{\alpha, z}$), pour automatiser cette intuition qui nous dit qu'« en cas d'inconsistance, fournir une solution – non réalisable donc – avec une explication de son incohérence qui soit la plus petite possible en terme d'importance vis-à-vis des préférences. » ? Comment relaxer le système de contraintes $\mathcal{R}$ de manière à pouvoir identifier des solutions non réalisables certes, mais de bonne qualité pour $\{c_{\succcurlyeq}[\alpha, z]\}_{\alpha, z}$, et adéquates à la situation d'incohérence ?

Avant toute investigation, les concepteurs et utilisateurs sont en droit de s'interroger sur la signification des contraintes dures de leur système de contraintes basé sur les préférences, et leurs spécificités par rapport aux contraintes souples. Nous avons vu au § 3.2.3.5 que la faisabilité d'une instance n'avait pas toujours le même sens profond que la faisabilité du système réel étudié. Certaines contraintes dures véhiculent ce sens. Mais qu'en est-il des autres ? Lorsqu'elles expriment des niveaux d'aspiration ou des niveaux de qualité minimale, il est alors légitime de pouvoir les relaxer. Cette réflexion permettra de mieux comprendre les rela-

tions entre les différentes contraintes dures, et de rechercher les liens entre contraintes dures et contraintes souples. Des préférences en termes de relaxation pourront alors résulter de cette réflexion, ainsi que des relations de dépendances entre contraintes dures et souples.

En présence de préférences, un autre problème se pose. Pour des relaxations équivalentes, il est nécessaire de comparer les sous-ensembles de solutions réalisables des systèmes de contraintes relaxées pour choisir l'explication. Prenons un exemple illustré par la Figure 23 :

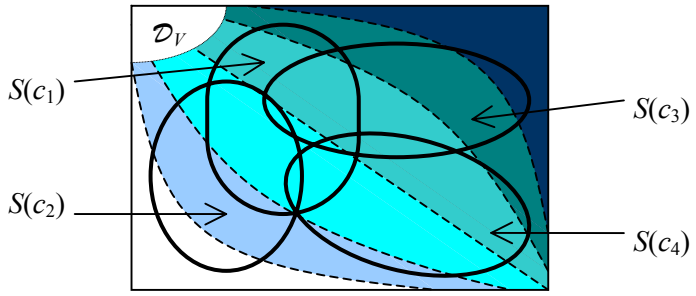


Figure 23 : Illustration de l'incohérence sur un système de contraintes dont les solutions sont totalement préordonnées.

Légende :
- Le rectangle schématise l'ensemble des solutions $\mathcal{D}_V$.
- Chaque disque désigne l'ensemble des solutions réalisables d'une contrainte.
- Le dégradé exprime les préférences : Plus la couleur est foncée, plus la valeur de la solution est élevée.

Considérons un P-CS ($\mathcal{K} = (V, D, C)$, $\{c_{\succcurlyeq}[\alpha, z]\}_{\alpha, z}$) avec :
$C = \{c_1, c_2, c_3, c_4\}$; $\succcurlyeq$ est un préordre total ; $\mathcal{K}$inconsistant $\Leftrightarrow S_{CSP}(\mathcal{K}) = \varnothing$, et $S_{CSP}((V, D, C_i)) \neq \varnothing$ en notant $C_i = C \setminus \{c_i\}$ pour tout $i = 1, \ldots, 4$. Supposons en plus qu'en termes de relaxation, moins il y a de contraintes supprimées, plus la relaxation est interessante. De plus, supprimer c_i est équivalent à supprimer c_j pour tout $1 \leq i, j \leq 4$. En regardant la Figure 23, la meilleure relaxation serait celle qui supprime la contrainte c_2, car $S_{CSP}((V, D, C_2))$ propose des solutions de meilleure qualité que $S_{CSP}((V, D, C_i))$ pour tout $i \in \{1, 3, 4\}$.

Ainsi, dans le cadre d'un préordre total $\succcurlyeq$, un problème de relaxation cohérente, tenant compte de $\succcurlyeq$, devrait retourner C_2. Pour pouvoir identifier la meilleure relaxation entre plusieurs relaxations équivalentes, le problème se base sur le principe suivant :

$\forall A, B \subseteq \mathcal{D}_V$, A est préféré à B $\Leftrightarrow$ les solutions de meilleure qualité pour A sont aussi les solutions de meilleure qualité pour $A \cup B$.

A est indifférent à B $\Leftrightarrow$ les solutions de meilleure qualité pour A sont équivalentes aux solutions de meilleure qualité pour B.

La recherche d'explication, à partir d'une relation binaire de préférence $\succcurlyeq$, est donc basée sur la comparaison de sous-ensembles de $\mathcal{D}_V$.

Mais dans le cas général, est-il toujours possible de comparer les meilleures solutions de deux sous-ensembles non vides A et B de $\mathcal{D}_V$? Si la réponse est négative, y a-t-il des conditions, un cadre ? Dans le cas transitif complet, les axiomes de comparaison d'ensembles sous-jacents à l'optimisation consistent à comparer les sous-ensembles via leurs meilleurs éléments. La relation binaire définie sur $\mathcal{P}(S)$ par le principe ci-dessus est un préordre total.

Hors du cadre transitif, ces axiomes ne sont plus immédiats. Identifions le problème par un exemple : La Figure 24 présente le cas de deux ensembles $A = \{x, y, z\}$ et $B = \{x, y, t\}$ possédant chacun un ensemble maximal : $M(A, \succcurlyeq) = \{x\}$, $M(B, \succcurlyeq) = \{y\}$, mais où leur union ne possède pas d'ensemble maximal : $M(A \cup B, \succcurlyeq) = \varnothing$.

Remarque : les arcs réflexifs ont été omis.

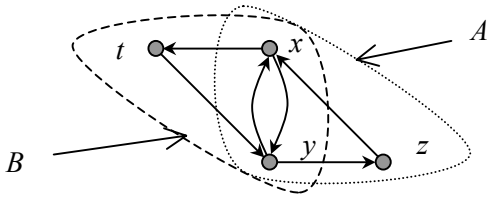


Figure 24 : Illustration de difficultés rencontrées lorsque deux ensembles maximaux sont comparés.

Des cas d'école peuvent aussi être construit, où l'union des ensembles maximaux de A et B n'a pas d'ensemble maximal. Par exemple, pour $A = \{x, y\}$ et $B = \{z, t\}$, nous avons $M(M(A, \succcurlyeq) \cup M(B, \succcurlyeq), \succcurlyeq) = \varnothing$. A moins d'utiliser des axiomes de choix identifiant pour tout sous-ensemble A non vide un ensemble de choix non vide, comme les e.r.o.m.i. ou les e.r.m.m.i. (§ 3.4.4), il n'existe pas toujours un ensemble de choix pour une axiomatique donnée. Dans ce cas, pour envisager d'utiliser une axiomatique de choix SA, il est nécessaire d'identifier la structure minimale pour $\succcurlyeq$ telle que $C_{SA}(A) \neq \varnothing \;\; \forall\, A \subseteq \mathcal{D}_V$, et de s'y restreindre.

Les axiomes de comparaison d'ensembles permettent de répondre aux questions suivantes :

– Pour tout sous-ensemble A, B non vide de $\mathcal{D}_V$, est-ce que pour tout élément x de B il existe un élément y de A vérifiant que $y \succcurlyeq x$?
– Est-ce que pour tout élément x de $C_{SA}(B)$, il existe un élément y de $C_{SA}(A)$ vérifiant que $y \succcurlyeq x$?

Plus généralement,

– Est-ce que l'on compare A et B ? $C_{SA}(A)$ et $C_{SA}(B)$? Est-ce la même chose ? Dans quels cas est-ce la même chose ? Comment la comparaison se fait-elle ? Élément par élément ? Sous-ensemble par sous-ensemble ? En évaluant où se trouvent les meilleurs éléments de l'union $C_{SA}(A \cup B)$ ou $C_{SA}(C_{SA}(A) \cup C_{SA}(B))$?

Cette comparaison aboutit à une relation binaire de préférence sur $\mathcal{P}(S)$.

5.3.4.2. Rationalisabilité et Aide face à l'Incohérence en Situation de Choix

La comparaison des ensembles de choix de sous-ensembles d'un ensemble structuré par une relation binaire de préférence $(S, \succcurlyeq)$ donné, a fait l'objet de beaucoup d'études en TCS[120]. Les travaux ont porté principalement sur l'ensemble optimal, et depuis peu sur l'ensemble maximal (SUZUMURA [1983], SEN [1997]).

Un des problèmes étudié est le suivant : Ils cherchent à caractériser les relations non binaires de préférence $rnp(S) = (\mathcal{A}, C)$, pour lesquelles il existe une relation binaire de préférence $\succcurlyeq_{rnp}$ dont l'ensemble optimal (resp. maximal) en chaque sous-ensemble $A \in \mathcal{A}$ est égal à $C(A)$: pour tout $A \in \mathcal{A}$, $B(A, \succcurlyeq_{rnp}) = C(A)$ (resp. : $M(A, \succcurlyeq_{rnp}) = C(A)$). Une telle relation non binaire de préférence est dite **rationalisable**. La relation binaire $\succcurlyeq_{rnp}$ est obtenue, via la relation non binaire $rnp(S)$ de différentes façons appelées **axiomes des préférences révélées**. En voici deux :

$$\succcurlyeq_{rnp} = \{(x, y) \in C(A) \times A, \text{ pour tout } A \in \mathcal{A}\}, \text{ et}$$
$$\succcurlyeq_{rnp} = \{(x, y) \in C(A) \times (A \setminus C(A)), \text{ pour tout } A \in \mathcal{A}\}\ .$$

Notre problème suppose donnée une relation binaire de préférence $\succcurlyeq$, une axiomatique de choix SA, et considère la relation non binaire de préférence associée $rnp_{SA,\,\succcurlyeq}(S)$ obtenue via la formule (3–13) donnée au § 3.4.6.3. Nous cherchons alors à savoir, (*a*) si cette relation non binaire de préférence est complète (§ 3.3.4), et (*b*) si les opérations ensemblistes, comme l'inclusion, l'union et l'intersection induisent des propriétés particulières sur les ensembles de

[120] Ces études ont été menées dans le cadre des relations non binaires de préférence et des fonctions de choix (§ 3.3.4). Nous renvoyons à toute la riche littérature sur le sujet : SEN [1970, chap. 1*], [1971], [1986], [1993*b*], [1997*a*], SUZUMURA [1983, chap. 2], parmi d'autres.

choix. Notre but à terme, est de savoir quand il est possible et acceptable de comparer deux ensembles de choix de deux sous-ensembles non vides A, B de S.

La TCS a établi des conditions nécessaires et suffisantes sur $\succcurlyeq$ pour que la relation non binaire de préférence $rnp_{SA,\succcurlyeq}(S)$ soit complète, lorsque SA est l'optimalité (resp. la maximalité). Ainsi, nous avons les propriétés suivantes (SEN [1970], SUZUMURA [1983]) :

(a) Il existe un ensemble optimal pour tout sous-ensemble non vide de $(S, \succcurlyeq)$ ssi $\succcurlyeq$ est P-acyclique complet.

(b) Il existe un ensemble maximal pour tout sous-ensemble non vide de $(S, \succcurlyeq)$ ssi $\succcurlyeq$ est fortement P-acyclique.

Plusieurs autres résultats et axiomes pertinents sont disponibles en TCS. Nous ne nous investirons pas ici.

Au delà de ces questions de comparaison d'ensembles, d'autres problèmes doivent être résolus avant d'arriver à une problématique technique ayant des bases solides, et à partir desquelles des algorithmes pourront émerger.

5.3.5. Choix, Consistance et Systèmes de Contraintes Basés sur les Préférences

Le choix peut aussi être couplé à la consistance, en partant de l'idée que pour un système de contraintes basé sur les préférences, seules les solutions choisies sont pertinentes. Cette problématique mixte d'aide à la décision consiste alors à supprimer les valeurs et/ou combinaisons de valeurs qui n'aboutissent à aucune solution de meilleure qualité.

En s'inspirant des processus de décision basés uniquement sur la consistance pour résoudre de manière interactive des problèmes décisionnels (§ 3.2.3.4), cette problématique mixte peut être utilisée à des fins similaires. L'aide à la décision consiste alors à décrire au mieux les valeurs (ou combinaison de valeurs) pouvant conduire à une solution de meilleure qualité, les décideurs effectuant seuls les choix pertinents.

Nous parlerons de **consistance qualitative**, pour désigner cette problématique mixte d'aide à la décision. Les problèmes de consistance qualitative font usage à la fois d'axiomes de choix et d'axiomes de consistance. Ils font aussi appel à d'autres axiomes spécifiques appelés **axiomes de consistance qualitative**. Ces axiomes ont aussi des liens très forts avec la TCS comme nous allons voir tout de suite.

Un système de contraintes basé sur les préférences $(\mathcal{R}_1 = (V, D_1, C_1), \{c_{\succcurlyeq 1}[\alpha, z]\}_{\alpha, z})$ est ***SA*-équivalent** à un autre système de contraintes basé sur les préférences $(\mathcal{R}_2 = (V, D_2, C_2), \{c_{\succcurlyeq 2}[\alpha, z]\}_{\alpha, z})$ si et seulement si un ensemble de choix pour $(S_{\mathrm{CSP}}(\mathcal{R}_1), \succcurlyeq_1)$ satisfaisant l'axiomatique SA est aussi un ensemble de choix pour $(S_{\mathrm{CSP}}(\mathcal{R}_2), \succcurlyeq_2)$ satisfaisant cette même axiomatique SA ; et vice et versa. En d'autres termes, pour tout $C_{SA}(S_{\mathrm{CSP}}(\mathcal{R}_1), \succcurlyeq_1)$, il existe $C_{SA}(S_{\mathrm{CSP}}(\mathcal{R}_2), \succcurlyeq_2)$ tel que : $C_{SA}(S_{\mathrm{CSP}}(\mathcal{R}_1), \succcurlyeq_1) = C_{SA}(S_{\mathrm{CSP}}(\mathcal{R}_2), \succcurlyeq_2)$; et vice et versa.
Cette relation d'équivalence sur les systèmes de contraintes basés sur les préférences, garantit la conservation des solutions choisies selon l'axiomatique de choix SA. Par exemple, $(\mathcal{R}_1, \{c_{\succcurlyeq 1}[\alpha, z]\}_{\alpha, z})$ est **optimal-équivalent** à un autre système de contraintes basé sur les préférences $(\mathcal{R}_2, \{c_{\succcurlyeq 2}[\alpha, z]\}_{\alpha, z})$ si et seulement si ces deux systèmes ont même ensemble optimal.

Le problème usant de préférences et équivalent au GIC consiste à supprimer les valeurs de domaines qui ne conduisent à aucune solution appartenant à un ensemble de choix dont l'axiomatique est donné. En termes formels, nous obtenons :

SA-GIC : Étant donné un système de contraintes basé sur les préférences ($\mathcal{R}_1 = (V, D_1, C_1)$, $\{c_{\succcurlyeq 1}[\alpha, z]\}_{\alpha, z}$), et une axiomatique de choix *SA*, retourner un système de contraintes *SA*-équivalent ($\mathcal{R}_2 = (V, D_2, C_2)$, $\{c_{\succcurlyeq 2}[\alpha, z]\}_{\alpha, z}$) tel que :

$$\begin{cases} C_2 = C_1 \\ \succcurlyeq_1 = \succcurlyeq_2 \\ \varnothing \neq D_2(v) \subseteq D_1(v), \text{ pour tout } v \in V, \text{ et} \\ \text{pour tout } v \in V \text{ et} & \left| \begin{array}{l} x \in S_{\mathrm{CSP}}(\mathcal{R}_1) \\ x(v) = d \\ \text{il existe } C_{SA}(S_{\mathrm{CSP}}(\mathcal{R}_2), \succcurlyeq_2) \text{ avec } x \in C_{SA}(S_{\mathrm{CSP}}(\mathcal{R}_2), \succcurlyeq_2) \end{array} \right. \\ \text{pour tout } d \in D_2(v), & \\ \text{il existe } x \in S_{\mathrm{CSP}}(\mathcal{R}_2) \text{ tel que :} & \end{cases},$$

si un tel système existe. Sinon, retourner “non”.

Par exemple, l’optimalité-GIC permet de conserver uniquement les valeurs de domaines pour lesquelles il existe une solution optimale les utilisant. Lorsque la relation binaire de préférence $\succcurlyeq_1$ est acyclique, une valeur de domaine peut être éliminée s’il existe une autre valeur du même domaine utilisée par une solution réalisable de meilleure qualité que toutes les solutions réalisables utilisant la première valeur. Remarquons que ($\mathcal{R}_1$, $\{c_{\succcurlyeq 1}[\alpha, z]\}_{\alpha, z}$) est *SA*-équivalent à ($\mathcal{R}_2$, $\{c_{\succcurlyeq 2}[\alpha, z]\}_{\alpha, z}$). Le problème *SA*-GIC suppose que nous ayons accepté l’axiome de consistance qualitative suivant :

$$C_{SA}(C_{SA}(S_{\mathrm{CSP}}(\mathcal{R}_1), \succcurlyeq_1)) = C_{SA}(S_{\mathrm{CSP}}(\mathcal{R}_1), \succcurlyeq_1). \qquad \textbf{(axiome de rnp-stabilité)}$$

Cet axiome est une adaptation directe de l’axiome de stabilité utilisé en TCS[121] (SUZUMURA [1983, § 2.3.2]). Il exprime la conservation des éléments choisis. En d’autres termes, il n’y a aucune raison de choisir un élément plus qu’un autre lorsqu’on se limite aux éléments choisis au premier abord. Les couples (*SA*, $\succcurlyeq_1$) vérifiant cet axiome sont assez nombreux. Par exemple, lorsque l’axiomatique de choix *SA* correspond à l’optimalité, à la maximalité ou à la maximalité stable maximale, et que la relation $\succcurlyeq_1$ est transitive, quasi-transitive ou acyclique (voir SUZUMURA [1983]). Le problème de *SA*-GIC est relié au problème GIC par la relation suivante :

$$SA\text{-GIC}(\mathcal{R}, \{c_{\succcurlyeq}[\alpha, z]\}_{\alpha, z}) = \mathrm{GIC}(\mathcal{R}_2), \text{ avec } S_{\mathrm{CSP}}(\mathcal{R}_2) = \bigcup C_{SA}(S_{\mathrm{CSP}}(\mathcal{R}), \succcurlyeq). \qquad (5\text{–}28)$$

En d’autres termes, les solutions réalisables pour $\mathcal{R}_2$ sont les solutions de meilleure qualité pour $\mathcal{R}$.

Un autre résultat de la TCS peut être utilisé pour caractériser d’autres couples (*SA*, $\succcurlyeq_1$). Ainsi, supposons que nous ayons deux systèmes de contraintes basés sur les préférences ($\mathcal{R}_1 = (V, D_1, C_1)$, $\{c_{\succcurlyeq 1}[\alpha, z]\}_{\alpha, z}$) et ($\mathcal{R}_2 = (V, D_2, C_2)$, $\{c_{\succcurlyeq 2}[\alpha, z]\}_{\alpha, z}$) *SA*-équivalents, et vérifiant C_1

[121] L’axiome de stabilité utilisé ici est un axiome de consistance qualitative, à ne pas confondre avec l’axiome de stabilité (ou d’indépendance) vu au § 3.4.2 qui est un axiome de choix. Le premier est une propriété des relations non-binaires de préférence, alors que le second est une propriété des ensembles de choix. Pour éviter les confusions, l’axiome qui nous intéresse est noté rnp-stabilité, ‘rnp-’ étant mis pour ‘relation non-binaire de préférence’. L’axiome original est défini comme suit :

Étant donnée une relation non-binaire de préférence $rnp(S) = (\mathcal{A}, C)$ sur un ensemble non vide *S*. $rnp(S)$ vérifie l’**axiome de rnp-stabilité** si et seulement si : $\forall A \in \mathcal{A}, \quad C(A) \in \mathcal{A} \Rightarrow C(C(A)) = C(A)$

Pour effectuer le lien avec notre problème, rappelons qu’à partir de tout triplet (S, $\succcurlyeq$, *SA*), avec *S* un ensemble, $\succcurlyeq$ une relation binaire de préférence sur *S* et *SA* un système axiomatique de choix, il est possible de définir une relation non binaire de préférence en utilisant la définition (3–13) vu au § 3.4.6.3.

$= C_2$, $\succcurlyeq_1 = \succcurlyeq_2$, et $D_2(v) \subseteq D_1(v)$ pour tout $v \in V$. Alors l'axiome de Nash[122] (SUZUMURA [1983]) est vérifié par le quadruplet $(S_{\text{CSP}}(\mathcal{R}_1), S_{\text{CSP}}(\mathcal{R}_2), SA, \succcurlyeq_1)$:

$$[\ S_{\text{CSP}}(\mathcal{R}_2) \subseteq S_{\text{CSP}}(\mathcal{R}_1) \quad \text{et} \quad C_{SA}(S_{\text{CSP}}(\mathcal{R}_1), \succcurlyeq_1) \subseteq S_{\text{CSP}}(\mathcal{R}_2)\ ] \ \Rightarrow \quad C_{SA}(S_{\text{CSP}}(\mathcal{R}_1), \succcurlyeq_1) = C_{SA}(S_{\text{CSP}}(\mathcal{R}_2), \succcurlyeq_1)) \qquad \textbf{(axiome de Nash)}$$

En d'autres termes, si les solutions réalisables sont réduites en passant de $\mathcal{R}_1$ à $\mathcal{R}_2$, mais conservent les éléments choisis dans $\mathcal{R}_1$, alors les éléments choisis dans $\mathcal{R}_2$ sont les mêmes que ceux choisis dans $\mathcal{R}_1$. Ces systèmes de contraintes ne vérifient pas forcément la propriété (5–28). Par contre ils vérifient toujours l'axiome de rnp-stabilité (voir SUZUMURA [1983, théorème 2.12]). Cet axiome permet de caractériser les niveaux de consistance qualitatifs inférieurs à *SA*-GIC, vérifiant simplement qu'il y a conservation des éléments choisis sans que l'ensemble des solutions réalisables soient limitées à ces éléments.

En définitive, la problématique de la consistance qualitative consiste donc à décrire les solutions de meilleure qualité d'un système de contraintes basé sur les préférences. Elle est basée sur l'hypothèse bien fondée que seules les solutions de meilleure qualité intéressent nos décideurs. Remarquons que d'autres propriétés (axiomes) que la *SA*-équivalence peuvent être utilisés pour appuyer nos problèmes de consistance qualitative. Ainsi, au lieu d'utiliser l'union de tous les ensembles de choix de la relation (5–28), on peut utiliser l'intersection ou l'union des ensembles de choix de cardinalité minimale. Le domaine réalisable $S_{\text{CSP}}(\mathcal{R}_2)$ de $\mathcal{R}_2$ peut aussi correspondre exactement à un ensemble de choix de $(S_{\text{CSP}}(\mathcal{R}), \succcurlyeq)$, etc.

Du fait que les problèmes de consistance qualitative s'inscrivent dans une dynamique répétitive, il est souvent nécessaire d'asseoir la consistance qualitative sur des axiomes valables pour cette dynamique. Par exemple, si la dynamique consiste à ajouter des contraintes dures temporaires (procédurales), un axiome de conservation des ensembles de choix initiaux doit être satisfait. Plus précisément, considérons un P-CS initial $(\mathcal{R}, \{c_{\succcurlyeq}[\alpha, z]\}_{\alpha, z})$, un système de contraintes courant $\mathcal{R}_1$ obtenu à partir de $\mathcal{R}$ en instanciant par exemple des variables et supprimant des valeurs dans les domaines ; nous recherchons un système de contraintes $\mathcal{R}_2$ vérifiant :

$$S_{\text{CSP}}(\mathcal{R}_2) \subseteq \bigcup C_{SA}(S_{\text{CSP}}(\mathcal{R}), \succcurlyeq) \text{ et } \mathcal{R}_2 = \text{GIC}(\mathcal{R}_1)$$

Maintenant, si la rationalité est procédurale (les préférences des acteurs dépendent de la dynamique), cette condition n'est plus valable, et des travaux axiomatiques similaires à ceux de la section précédente doivent être effectués.

Il en est de même des exigences algorithmiques : Pour établir un algorithme pour la *SA*-GIC, il sera souvent nécessaire que, pour un P-CS $(\mathcal{R}, \{c_{\succcurlyeq}[\alpha, z]\}_{\alpha, z})$, le triplet $(\mathcal{D}_V, \succcurlyeq, SA)$ vérifie l'axiome de rnp-stabilité initial (voir note en bas de page numéro 121). Quelques fois, l'axiome de Nash initial (voir note en bas de page numéro 122) sera de rigueur.

---

[122] Tout comme l'axiome de rnp-stabilité, l'axiome de Nash est une propriété des relations non binaires de préférence. Il a été défini en TCS de manière légèrement différente, en considérant une relation non-binaire de préférence $rnp(S) = (\mathcal{A}, C)$ sur un ensemble non vide S. $rnp(S)$ vérifie **l'axiome de Nash** si et seulement si :

$$\forall\ A, B \in \mathcal{A}, \qquad [\ A \subseteq B \quad \text{et} \quad C(B) \subseteq A\ ] \ \Rightarrow \ C(A) = C(B)$$

[123] Contrairement aux contraintes dures qui n'admettent que des paramètres liés au domaine d'application.

6. L'Équité comme Outil d'Aide à l'Évaluation en Planification

6.1. Introduction

Un thème récurrent dans les problèmes d'élaboration de plannings de travail de personnes, est l'équité. Pour justifier ou au contraire démentir (réfuter) les modèles préférentiels utilisés, nous nous sommes penchés sur les études portant sur ce concept effectuées en sciences humaines et sociales.

L'équité tient une place de choix dans l'évaluation des plannings de personnel. Elle revêt une forme différente d'un modèle à l'autre (§ 2.3.5). Ainsi WARNER [1976], ou encore JAUMARD *et al*. [1998], la définissent comme un quota de points égal pour tous les employés tenant compte de la qualité des plannings précédents. Dans JASZKIEWICZ [1997] et [2001], l'équité est traduite par la répartition la plus équilibrée entre les infirmières du nombre de fois qu'un matin, un après-midi et un soir sont affectés. Souvent, un grand nombre de politiques (positionnement des repos, etc.) sont mises en place pour apporter plus d'équité dans les plannings, bien que le concept d'équité ne soit pas mentionné.

Nous avons décidé de fonder le modèle préférentiel des PEPTP en priorité sur des considérations axiomatiques, ceci, afin d'améliorer notre compréhension du problème[124]. Elle permettra aussi à terme de concevoir des modèles de préférences de meilleure qualité par rapport à l'existant, pour l'identification des solutions de meilleur compromis. Les deux principes sur lesquels nous nous appuyons sont l'efficacité et l'équité. L'équité, comme l'efficacité, est perçue comme une notion agrégative, dont le but est de gérer une partie des conflits d'intérêts entre individus.

Un des champs d'application de la théorie du choix social, est l'exploration de concepts moraux (philosophie morale, sociale) (voir § 3.1.2) comme l'équité. Nous étudierons ce concept en détail dans ce chapitre. Les axiomes associés à l'équité sont de plus en plus utilisés dans d'autres disciplines des SHS telles que la théorie des jeux, l'économie politique, la théorie de la justice, la philosophie (plus spécialement l'éthique et la philosophie sociale). Ils ne le sont pas uniquement dans les disciplines formelles. Ainsi, dans la pratique, plusieurs organismes internationaux (Fond Monétaire International (FMI), Bureau du Développement Humain de l'ONU, etc.) ont basé leurs indicateurs sur ces axiomes.

Dans ce chapitre nous utilisons principalement les préférences collectives structurées en relation binaire de préférence ou en vrai-critère. Nous disons qu'une solution est plus **équitable** (resp. **équitable et efficace**) qu'une autre, s'il existe des principes d'équité (resp. d'équité et d'efficacité) clairs, suffisants et non contradictoires en faveur de cette assertion. Dans le cas contraire, elles sont dites incomparables pour de tels arguments. Étant donnée une règle d'agrégation *RA* véhiculant des principes d'équité (resp. d'équité et d'efficacité), une solution x sera dite équitable (resp. équitable et efficace) pour l'instance (S, $pref(S)$) si x est une solution de meilleur compromis pour *RA*. En d'autres termes, x fait partie de l'ensemble de choix collectif pour *RA*, du couple (S, $pref(S)$).

[124] au lieu d'appliquer – ou se cacher derrière ! – une formule d'agrégation trop abstraite et souvent décevante.

Nous nous intéressons plus particulièrement aux règles d'agrégation jugeant deux solutions incomparables lorsqu'elles ne sont pas comparables pour des principes d'équité (resp. d'équité et d'efficacité) préalablement validées. Cette incomparabilité peut être exploitée pour élaborer des procédures interactives, laissant aux utilisateurs le soin de décider des cas litigieux non résorbés par ces principes pour résoudre des conflits globaux (§ 4.2.1.1).
Cet interêt est guidé par le leitmotiv de prudence lors de la modélisation et l'agrégation des préférences, traduit plus généralement par la citation de SEN [1992] rappelée en début de chapitre 3.

Grandes lignes du chapitre.

Le reste du chapitre est formé de trois sections. La première (§ 6.2) s'intéresse aux règles d'agrégation et aux axiomes véhiculant des principes d'équité et d'efficacité dans le cadre des problèmes d'allocation de ressources. Ainsi, la dominance de Lorenz et ses généralisations sont présentées, et l'accent est mis sur leurs relations avec la théorie polyédrale. Les problèmes d'allocation de ressources supposent généralement que les préférences individuelles sont des vrai-critères. La section 6.3 est consacrée à ces mêmes études, mais dans un contexte préférentiel plus pauvre : les préférences individuelles et les comparaisons interpersonnelles sont fournies par des relations binaires de préférence. Dans la dernière section (§ 6.4), nous mettons en pratique sommairement les résultats des sections précédentes aux problèmes d'élaboration de plannings de travail de personnel (PEPTP). Un modèle préférentiel très générique pour les PEPTP, basé sur l'équité et l'efficacité, est alors présenté.

6.2. Dominance de Lorenz et Généralisations pour l'Évaluation de l'Équité dans les Problèmes d'Allocation de Ressources

6.2.1. Un Cas d'École : Le Problème de Pure Répartition

Pour mieux identifier les difficultés liées à la mesure de l'équité, nous allons partir de l'exemple très simple qu'est le **problème de pure répartition** (*fair cake division problem*). Ce dernier considère une ressource, un bien en quantité finie fixe infiniment divisible, à distribuer aux individus d'une population. De plus, il est admis que chaque agent a un droit égal à n'importe quelle quantité de bien disponible, et que personne ne peut revendiquer un traitement spécial basé sur ses caractéristiques personnelles (D'ASPREMONT & GEVERS [2002, § 2.1]).
Par exemple, nous cherchons à répartir un gâteau homogène entre plusieurs personnes tel que :

- chaque personne a autant d'utilité[125] (satisfaction, plaisir) que sa part est grosse,
- sa part de gâteau est sa seule source d'utilité,
- la comparaison interpersonnelle se fait sur la base de ces quantités allouées.

La Figure 25 nous montre 3 manières possibles de découper un gâteau. Comment exprimer par des axiomes ce que notre intuition nous dit lorsque nous comparons ces répartitions, à savoir la configuration de découpe (*c*) est plus équitable que la configuration (*b*) qui elle-même est plus équitable que (*a*) ?

[125] L'**utilité individuelle** est l'un des plus vieux concepts de l'économie (voir l'utilitarisme en page 136). Elle a été sujette à plusieurs interprétations. Nous utiliserons la plus courante. Par définition, l'utilité d'une personne est la satisfaction, le plaisir qu'elle tire d'une situation (solution) si celle-ci est adoptée (voir par exemple SEN [1992, préface de la version française], ou SEN[1987]).

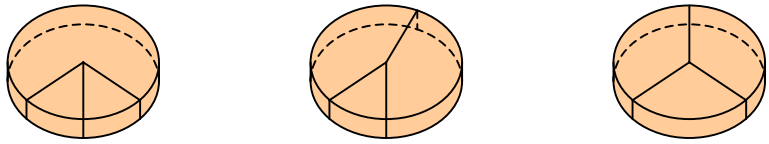


Figure 25 : Exemple de trois configurations de découpe d'un gâteau homogène.

Modélisation :

Une instance $I_{PR} = (E, Q)$ est formée d'une population E finie de n personnes ($\Leftrightarrow n = \mathrm{card}(E)$), et de la quantité totale Q de ressource à distribuer. Une distribution de la totalité de la ressource attribue une quantité $x(i)$ à chaque individu i, telle que la somme de ces quantités vaille Q, définissant ainsi l'ensemble des solutions réalisables du problème :

$$S(I_{PR}) = \{x \in \quad ^{n} \text{ tel que } x(i) \geq 0 \ \ \forall\, i \in \{1, \ldots, n\}, \ \text{ et } x(1) + \ldots + x(n) = Q\}$$

Les solutions réalisables sont les vecteurs de rationnels positifs de dimension n, dont la somme des composantes est égale à Q. Géométriquement, $S(I_{PR})$ est le simplexe de dimension $n - 1$, et la Figure 26 nous montre le cas $(n, Q) = (3, 12)$. Le simplexe de dimension 2 est aussi connu dans la littérature économique sous le nom de triangle de Kolm, ce dernier ayant été le premier à utiliser cette représentation comme support à l'intuition (voir par exemple ZHENG [2001] pour une utilisation pertinente dans la conception d'indicateurs de pauvreté).

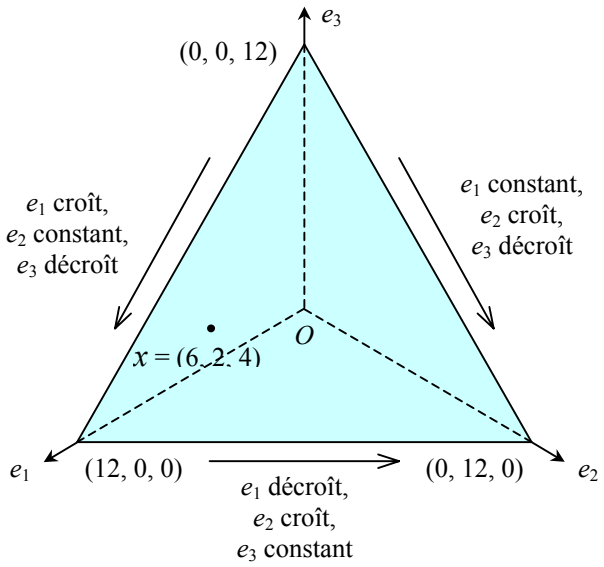


Figure 26 : Ensemble des solutions réalisables du problème de pure répartition entre trois individus $\{e_1, e_2, e_3\}$: simplexe de dimension 2.

Quels sont les problèmes à résoudre ?

Bien sûr, en tant qu'exemple d'école, la solution optimale x^* est celle qui donne la même quantité de biens à chaque individu de E, les hypothèses de départ stipulant qu'aucune information spécifique individuelle n'est prise en compte.

$$x^* = (\mu, \ldots, \mu) \in S(I_{PR}) \quad \text{avec } \mu = \frac{Q}{n} \tag{6-29}$$

Cependant, notre intérêt à étudier ce problème est de savoir ce qui arrive si nous remettons en question certaines hypothèses de l'énoncé. Donnons quelques exemples :

- Comment choisir la solution la plus équitable, lorsque x^* n'est pas dans l'ensemble des solutions réalisables (à cause par exemple d'ajout de contraintes linéaires ou d'intégrité, etc.) ;
- Comment identifier la plus équitable parmi deux solutions réalisables comparées ?
- De quelles informations devons-nous disposer pour définir un critère d'équité ?
- Comment faire lorsque la quantité totale est variable ? Ou dans le cas multi-ressources ?
- Comment définir l'équité,
  - Si les comparaisons interpersonnelles ne sont plus aussi évidentes et que nous voulons établir une solution de meilleur compromis ?
  - Si nous voulons tenir compte de caractéristiques personnelles, et des effets croisés ?
  - Lorsque l'utilité individuelle, dans une forme aussi restreinte, n'est plus le critère sur lequel sont basées les préférences individuelles, mais une utilité élargie, voire un autre critère comme le bien-être individuel ?
  - Lorsque d'autres critères, comme l'efficacité, doivent être pris en compte ? Et comment gérer les conflits générés ?

Dans la perspective d'une généralisation à des cas réels, comment répondre utilement aux questions précédentes ?

Nous allons analyser un peu mieux les hypothèses de départ.

Tout d'abord, les préférences individuelles : En se basant sur les hypothèses de départ (les préférences individuelles ne dépendent que de la quantité individuelle allouée), un profil simple $\mathcal{F}(E, S(I_{PR}))$ est construit en considérant un vrai-critère $g_i = (f_i, \geq)$ par individu i, la fonction de valeur étant définit par $f_i(x) = x(i)$ en la solution x.

Nous cherchons à caractériser les préférences collectives exprimant des jugements d'équité. Quels sont les axiomes que doivent respecter des préférences collectives pour qu'elles véhiculent des jugements équité ? Pour ce problème, nous nous limitons aux préférences sociales structurées en vrai-critère $g_{RA(\mathcal{F})}$ ou en relation binaire de préférence $\succcurlyeq_{RA(\mathcal{F})}$. Pour ne pas trop surcharger l'écriture, nous omettons le profil utilisé lorsque celui-ci va de soit : g_{RA} et $\succcurlyeq_{RA}$.

Que peut-on dire des préférences collectives ? La structure g_{RA} (resp. $\succcurlyeq_{RA}$) doit vérifier l'axiome d'égalité en proportions, lequel indique que si la solution réalisable x^* attribuant la même quantité de ressource à chacun existe, alors elle doit être choisie :

Axiome d'Égalité des Proportions (EP) : Étant donnée x^* définie par (6-29), g_{RA} (resp. $\succcurlyeq_{RA}$) vérifie :

$$g_{RA}(x^*) > g_{RA}(y) \quad (\text{resp. } x^* \succ_{RA} y) \quad \text{pour tout } y \in S(I_{PR}).$$

Une version faible est obtenue en remplaçant l'inégalité stricte ($>$ et $\succ_{RA}$) en inégalité faible ($\geq$ et $\succcurlyeq_{RA}$).

Un second axiome, reflétant bien les hypothèses de départ, est celle d'anonymat. Ce concept peut prendre différentes formes. Celle qui nous intéresse est l'invariance sous les permutations des quantités affectées :

Axiome d'Invariance sous les Permutations (IP) : Pour toute permutation σ des individus de E, et pour toute solution $x \in S(I_{PR})$, g_{RA} (resp. $\succcurlyeq_{RA}$) vérifie :

$$g_{RA}(x) = g_{RA}(y) \quad (\text{resp. } x \simeq_{RA} y) \quad \text{avec } y(i) = x(\sigma(i)), \text{ pour tout } i \in E.$$

Le partage de la ressource étant fait, quelque soit l'affectation aux personnes de E choisie, celle-ci laisse les préférences sociales indifférentes.

Un dernier axiome reflétant des considérations égalitaires très attrayantes est le principe de transfert de Pigou-Dalton (SEN [1973], MARSHALL & OLKIN [1979]). Cet axiome a été présenté pour la première fois indépendamment par PIGOU et DALTON au début du XXe siècle,

sous une forme, il est vrai, plutôt informelle, et dans le contexte des revenus. Il indique que (Figure 27) tout transfert d'une personne plus riche vers une plus pauvre, toute chose étant égale par ailleurs, devrait diminuer l'inégalité (et donc augmenter les préférences sociales).

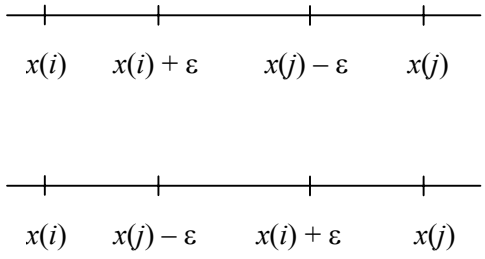


Figure 27 : Illustration du principe de transfert de Pigou-Dalton

Principe de Transfert de Pigou-Dalton (PDT) : Pour tout couple d'individus $(i, j) \in E \times E$, pour toutes solutions x et $y \in S(I_{PR})$ et pour tout réel $\varepsilon > 0$, g_{RA} (resp. $\succcurlyeq_{RA}$) vérifie :

$$\left.\begin{array}{l} y(i) = x(i) + \varepsilon < x(j) \\ y(j) = x(j) - \varepsilon > x(i) \\ y(k) = x(k) \quad \text{si } k \neq i, j \end{array}\right\} \quad \Rightarrow \quad g_{RA}(y) > g_{RA}(x) \quad (\text{resp. } y \succ_{RA} x)$$

Munis de ces informations, nous allons étudier les possibilités (avantages et inconvénients) offertes par les outils usuels de la RO (modèles linéaires, la règle de Pareto), ceux de l'AMCD (les règles basées sur le principe de concordance et de discordance), et ceux de l'économie expérimentale (indicateurs statistiques) débouchant sur des modèles non linéaires.

6.2.2. Quelques Outils Usuels de Mesure des Inégalités

6.2.2.1. Utilisation d'un Critère Collectif Linéaire

Étudions ce que nous gagnons et perdons en utilisant une règle d'agrégation de type somme pondérée des vrai-critères individuels $x(i)$. L'ensemble des solutions réalisables étant un polytope, l'utilisation d'une telle fonction d'agrégation linéaire permet de définir un problème d'optimisation (programme) linéaire :

$$\text{Maximiser } \sum_{i \in E} w(i) \cdot x(i),$$

$$\text{sous les contraintes : } \begin{cases} \sum_{i \in E} x(i) = Q \\ x(i) \geq 0 \quad \text{pour tout } i \in E. \end{cases} \tag{6-30}$$

Maintenant il reste à définir les poids $w(i)$. Est-il possible de les choisir de manière à respecter l'axiome d'égalité des proportions ? Malheureusement non. En effet, géométriquement, maximiser (ou minimiser) une fonction linéaire sur un polytope (non vide), aboutit toujours à des solutions optimales situées sur une face – à la frontière donc – de ce polytope[126] et jamais à l'intérieur ; Sauf dans le cas où l'ensemble des solutions optimales est le polytope tout entier[127]. Ce dernier cas arrive lorsque tous les poids sont égaux (la moyenne par exemple), et satisfait la version faible de l'axiome d'égalité des proportions. Dans le premier cas, aucun

[126] voir théorème 6 du chapitre 10 de SAKAROVITCH [1984*a*].

[127] Cela arrive lorsque la matrice formée du vecteur des poids de la fonction objective et de la matrice du système d'égalités implicites du polytope est de même rang que la matrice du système d'égalités implicites du polytope.

des deux axiomes n'est respecté, une solution optimale affectant toute la ressource à l'individu ou la coalition ayant le poids le plus élevé, laissant donc les autres sans rien[128].
Donc, le mieux que nous puissions trouver, c'est une fonction objective linéaire satisfaisant l'axiome d'égalité des proportions faible.

Dans les problèmes de planification (voir PARTOUCHE [1998]), le critère d'équité a été modélisé d'une autre manière. Au système de contraintes (6-30), utilisant des poids égaux à un[129], des contraintes linéaires paramétrés sont ajoutées, garantissant qu'un niveau d'inégalité maximale ne sera pas dépassé pour chaque individu :

$w(i) = 1$ pour tout $i \in E$

$x(i) \geq \beta(i)$ pour tout $i \in E$ $\beta(i)$ étant un paramètre à spécifier.

En d'autres termes, un **niveau de satisfaction**, ou niveau de qualité minimale, est assuré. La recherche d'une meilleure solution s'effectue alors par des interactions avec l'utilisateur (voir LÉVINE & POMEROL [1989, p. 104][130]). L'aide du décideur semble nécessaire. N'est il pas possible de mieux appréhender la modélisation et réduire les interactions ? Sinon, utiliser d'autres moyens ? Abandonnons momentanément cette alternative linéaire (nous y reviendrons au § 7.2) et tournons-nous vers la dominance de Pareto.

6.2.2.2. Utilisation de la Dominance de Pareto et de la Procédure Lexicographique

La règle de Pareto et la règle lexicographique forment à elles deux l'essentiel des règles non utilitaristes utilisées en programmation multiobjectif[131].

Le profil simple $\mathcal{F}(E, S(I_{PR}))$ peut être agrégé en utilisant la règle de Pareto. Cependant, elle ne respecte ni l'axiome d'égalité des proportions, ni la version faible : Cette règle d'agrégation érige toutes les solutions réalisables du problème de pure répartition en solutions maximales, efficaces (incomparables deux à deux). La recherche d'une meilleure solution parmi les solutions efficaces est laissée au décideur, via une exploration interactive et à l'utilisation de niveaux de satisfaction (voir note en base de page 130). Mais nous aboutissons à la même conclusion que le paragraphe précédent.

La règle lexicographique, elle, conduit à affecter la totalité de la ressource à un seul individu. L'utilisation de niveaux de satisfaction et d'une procédure interactive peut conduire à une solution moins extrémiste.

Et, qu'en est-il de la programmation mathématique ? Nous y reviendrons au § 7.2.

D'autres procédures comme la majorité, symbole généralement de la démocratie, ne donnent pas de résultats satisfaisants, comme nous le verrons tout de suite.

6.2.2.3. Utilisation de la Règle de la Majorité Généralisée

---

[128] Remarquons que lorsque certains poids sont négatifs ou nuls, les individus concernés ne se verront rien affecter, dans les solutions optimales ; Ne rien leur donner étant toujours plus efficace pour la population, dont les préférences correspondent à la somme pondérée à maximiser.

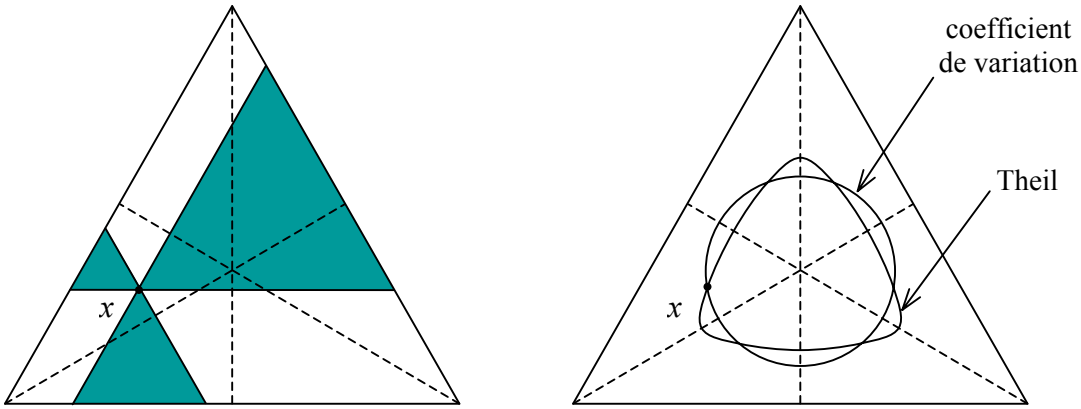
[129] Bien que la fonction objective ne soit plus nécessaire, car les poids égaux impliquent l'égalité entre l'ensemble des solutions optimales et l'ensemble des solutions réalisables, elle garde son utilité dans le cas où la quantité totale de ressources varie.

[130] « Le décideur fixe un plancher de satisfaction qu'il veut obtenir et il ne considère que les actions qui sont au-dessus du plancher. Puis il modifie ce niveau et procède par éliminations successives, à moins qu'une modification trop hardie n'ait entraîné l'absence de solution « au dessus », le décideur revient alors à des prétentions plus modestes et le processus continue ».

[131] voir EMELICHEV & PEREPELITSA [1994], ULUNGU & TEGHEM [1994], EHRGOTT & GANDIBLEUX [2000], parmi d'autres.

Traditionnellement, le concept de « démocratie » a souvent été identifié ni plus ni moins au gouvernement des individus par la majorité[132]. Ce type de règle d'agrégation peut-il conduire à une évaluation satisfaisante du problème de pure répartition ?

Rappelons (§ 4.2.1.3) que la règle de la majorité $\succcurlyeq_M$ consiste à préférer pour la collectivité, la solution qui est préférée par la majorité des individus. A partir d'une instance $I_{PR} = (E, Q)$ de problème de pure répartition on définit le profil $\mathcal{F}$ de vrai-critères $g_i = (f_i, \geq)\ \forall\ i \in E$ avec $f_i(x) = x(i)$ pour tout $x \in S(I_{PR})$. La règle de la majorité peut être utilisée sur $\mathcal{F}$. La Figure 28(*a*) montre le découpage de l'ensemble des solutions réalisables effectué par cette règle d'agrégation sur l'exemple du début (Figure 26) en la solution $x = (6, 2, 4)$.



(*a*) Découpe du domaine réalisable effectuée par la règle de la majorité. Les parties sombres représentent les solutions préférées à *x* pour la majorité.

(*b*) Courbes de niveau passant par *x* du coefficient de variation et de la mesure d'entropie de Theil.

Figure 28 : Quelques critères d'agrégation collective pour le problème de pure répartition.

En termes d'ensemble de choix, aucune solution réalisable du problème de pure répartition n'est optimale car toute solution réalisable fait partie d'un cycle de longueur 3 (corollaire du la proposition ci-dessous).

Proposition 10 : Considérons une instance de problème de pure répartition $I_{PR} = (E, Q)$, avec $\mathrm{card}(E) = n > 2$. Alors, pour tout $x \in S(I_{PR})$, il existe une solution $y \in S(I_{PR})$ vérifiant $y \succ_M x$.

Preuve : Voir Annexe B2.

Nous pouvons aller plus loin en montrant que l'ensemble des solutions réalisables pour le problème de pure répartition $S(I_{PR})$ n'a pas de noyau, ni d'ensemble maximal, et que cet ensemble est un e.r.o.m.i. et un e.r.m.m.i. (§ 3.4.4). Par conséquent, sur la base de l'optimalité relative ou de la maximalité relative, toute solution réalisable est une solution de meilleure qualité pour la majorité.

Tout ceci montre que la règle de la majorité ne respecte ni l'axiome d'invariance sous les permutations, ni le principe de transfert de Pigou-Dalton. Elle est plutôt inadaptée à l'identification des distributions équitables. Les notions de 'pouvoir' et de 'force' d'une coalition ne sont pas des concepts à mettre en avant pour traduire un principe moral comme l'équité.

Pour des références et une discussion globale sur l'utilisation de la règle de la majorité dans le contexte de l'allocation de ressources, nous renvoyons à SEN [1986, § 8.4].

Pourquoi pas les critères statistiques ?

[132] SUZUMURA [1983, chapitre 4] nous donne quelques références sur, selon ses propres termes, « la forte association doctrinale » qui a été faite entre la démocratie et la règle de la majorité.

6.2.2.4. Utilisation de Critères Statistiques[133]

Il existe un nombre impressionnant d'indicateurs statistiques, allant de la moyenne, à la mesure d'entropie de Theil, en passant par la variance, le coefficient de Gini et la déviation standard du logarithme. En fait, il est possible d'en construire une infinité comme nous verrons plus tard.

Tout d'abord un tri peut être fait en éliminant les indicateurs linéaires (moyenne, etc.). Concentrons-nous sur les indicateurs de dispersion. Nous étudierons plus en détail les quatre plus utilisées : le coefficient de variation (normalisation de la variance (indépendance vis-à-vis de la moyenne)), la déviation du logarithme, le coefficient de Gini et la mesure d'entropie de Theil (utilisé en théorie de l'information).

$$CV(x) = \frac{1}{\mu}\left(\frac{\sum_{i=1}^{n}(\mu - x(i))^2}{n}\right)^{1/2}$$

Coefficient de variation

$$H(x) = \left(\frac{\sum_{i=1}^{n}(\log(\mu) - \log(x(i)))^2}{n}\right)^{1/2}$$

Déviation standard du Logarithme

$$G(x) = \frac{1}{2n^2\mu}\sum_{i=1}^{n}\sum_{j=1}^{n}|x(i) - x(j)|$$

Coefficient de Gini

$$T(x) = \sum_{i=1}^{n} x(i)\cdot\log(n\cdot x(i))$$

Mesure d'entropie de Theil

avec μ la moyenne, qui correspond à la constante $Q \setminus n$ pour tout $x \in S(I_{PR})$.

La Figure 28 (*b*) montre sur le triangle de Kolm, les courbes de niveaux passant par $x = (6, 2, 4)$ du coefficient de variation[134] et de la mesure de Theil.

Elles semblent aller exactement dans le sens de notre intuition. Tout d'abord, elles satisfont l'axiome d'égalité des proportions. De plus, l'inégalité augmente avec l'indicateur, si l'on prend comme référence la valeur de x^* pour cet indicateur. En d'autres termes, si f_{stat} est un indicateur statistique de dispersion pris parmi les 4 présentés, pour toute solution x de $S(I_{PR})$ donnée, $f_{\text{stat}}(x) \geq f_{\text{stat}}(x^*)$. Dans l'exemple de la Figure 28 (*b*), les solutions jugées par l'indicateur plus équitables que la solution x, sont celles intérieures à la courbe de niveau. Donc en prenant comme fonction objective collective, l'inverse d'un des critères statistiques : $g_{RA} = (-f_{\text{stat}}, \geq)$, nous obtenons un outil proche de notre idée d'équité dans le cadre du problème de pure répartition.

Deux questions subsidiaires restent. Comment choisir notre indicateur ? Qu'est-ce qui les différencient ou les unissent ?

La première question, peut être vue comme une conséquence de la seconde : Une meilleure connaissance des indicateurs permettra d'effectuer un choix plus judicieux. Maintenant, si cette connaissance n'est pas facile à acquérir, une autre solution (SEN [1973]) est d'interroger chaque décideur sur le ou les indicateurs qu'il préfère, et d'en déduire une relation binaire de préférence collective $\succcurlyeq_{RA}$ en prenant l'intersection des indicateurs[135].

La seconde question a aussi conduit à plusieurs travaux, amorcés par Amartya SEN dans son livre sur l'inégalité (SEN [1973]), marquant par la même occasion le début de l'utilisation de la méthode axiomatique dans la conception d'indicateurs de qualité en économie. Nous en ferons le sujet principal du reste du chapitre.

6.2.3. La Dominance de Lorenz

[133] Cette section s'inspire du chapitre 2 de SEN [1973].

[134] Le problème de pure répartition tel qu'il est donné avec l'axiome d'égalité des proportions, suggère un vrai-critère qui minimise la distance (euclidienne par exemple) entre x et le point « idéal » x^*. Dans cet ordre d'idées, le coefficient de variation correspond (à un facteur multiplicatif près) à la distance euclidienne. Nous obtenons alors un programme quadratique qui peut être résolu en utilisant les outils de la programmation mathématique.

[135] L'« intersection d'indicateurs » correspond à l'agrégation de ces vrai-critères avec la dominance de Pareto.

6.2.3.1. Dominance de Lorenz comme Règle d'Agrégation pour le Problème de Pure Répartition

La Dominance de Lorenz (SEN [1973]) est l'un des critères les plus utilisés aujourd'hui comme outil de mesure de l'inégalité, car elle respecte les trois axiomes précités au § 6.2.1, et possède des propriétés communes à une majorité d'indicateurs statistiques acceptables.

La dominance de Lorenz est définie à partir de la courbe de Lorenz associée à une distribution.

La **courbe de Lorenz**[136] L_x d'une distribution x représente le pourcentage de la population rangée du plus pauvre en ressource au plus riche en fonction du pourcentage de ressource cumulé de ce pourcentage de population. La Figure 29 montre quatre courbes de Lorenz continues.

Pour tout $x \in S(I_{PR})$, vérifiant :

$$x(i) \leq x(i+1), \text{ pour tout } 0 < i \leq n-1$$

(après un tri par ordre croissant des composantes de x si nécessaire), la courbe de Lorenz associée est :

$$L_x(i/n) = \frac{1}{Q}\sum_{1<k\leq i} x(k) \quad \text{pour tout } i \in \{1, \ldots, n\}$$

avec $L_x(0) = 0$.

Cette courbe est discrète. Pour obtenir une courbe continue, nous pouvons utiliser la courbe en escaliers associée (Figure 31(b)) :

$$L_x(\lambda) = \frac{1}{Q}\sum_{i\leq n\cdot\lambda} x(i) \quad \text{pour tout } \lambda \in [0, 1]$$

ou une interpolation linéaire ou polynomiale.

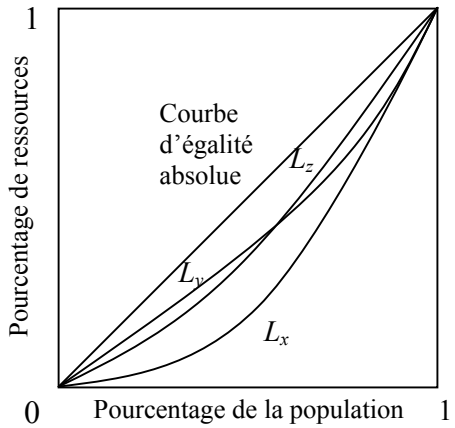


Figure 29 : Illustration de courbes de Lorenz.

La **dominance de Lorenz** $\succcurlyeq_L$ sur un ensemble S à partir des vrai-critères $g_i = (f_i, \geq)$ est une règle d'agrégation définie par :

$$x \succcurlyeq_L y \Leftrightarrow [\forall\, i : L_{f(x)}(i/n) \geq L_{f(y)}(i/n)],$$

avec $f(x)$ le vecteur de ^{n} dont les composantes sont $f_i(x)$. Cette relation définit entièrement les situations fondamentales d'indifférence, de préférence stricte et d'incomparabilité :

$$x \simeq_L y \Leftrightarrow x \succcurlyeq_L y \text{ et } y \succcurlyeq_L x.$$
$$x \succ_L y \Leftrightarrow x \succcurlyeq_L y \text{ et non } [y \succcurlyeq_L x].$$
$$x \parallel_L y \Leftrightarrow \text{non } [x \succcurlyeq_L y] \text{ et non } [y \succcurlyeq_L x].$$

La dominance de Lorenz est un préordre partiel sur S (conservation de la transitivité). La proposition « $x \succ_L y$ » se lit alors « x Lorenz-domine y » ou « y est Lorenz-dominé par x ».
Prenons l'exemple du problème de pure répartition. Considérons trois solutions réalisables x, y, z telles que leurs courbes de Lorenz correspondent respectivement à L_x, L_y et L_z de la Figure 29. Nous pouvons émettre une comparaison de Lorenz, en comparant simplement les courbes de Lorenz associées : x est Lorenz-dominé par y et z, car L_x est entièrement en dessous de L_y et L_z. Par contre y et z sont incomparables selon la dominance de Lorenz, car L_y et L_z se coupent. Remarquons que la ligne d'égalité absolue est la courbe de Lorenz associée à la solution x^*.

Il existe dans la littérature spécialisée, plusieurs caractérisations de cette règle d'agrégation. Ainsi, après quelques définitions préalables portant sur des polytopes associés

[136] Cette courbe est apparue pour la première fois dans LORENZ M. O. [1905] – Methods for Measuring Concentration of Wealth – *Journal of the American Statistical Association* 9.

aux permutations, nous présentons le théorème de caractérisation de Hardy, Littlewood et Polyà.

6.2.3.2. Polytopes, Permutation et Schur-concavité

Faisons une petite parenthèse sur un polytope très particulier qu'est le permutoèdre.

Une **matrice** carrée $Q = (q_{ij})$ de dimension $n \times n$ **doublement stochastique** (ou **bistochastique**) est une matrice à coefficients positifs ; dont chacune de ces lignes et colonnes sont formées de coefficients dont la somme vaut 1 :

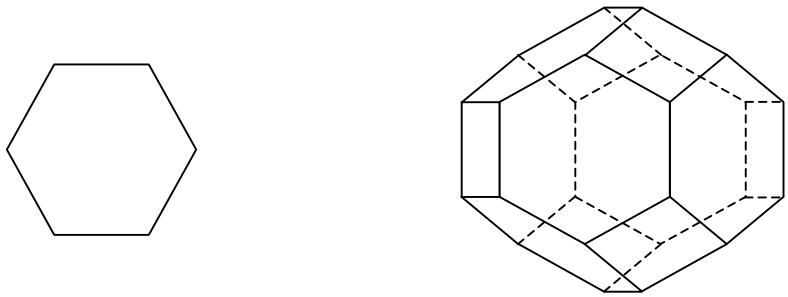
$$q_{ij} \geq 0 \text{ pour } 1 \leq i, j \leq n, \quad \sum_{j=1}^{n} q_{ij} = 1 \text{ pour } i = 1, \ldots, n \quad \text{et} \quad \sum_{i=1}^{n} q_{ij} = 1 \text{ pour } j = 1, \ldots, n.$$

Les **matrices de permutation** représentent un cas particulier de ces matrices. Elles comportent un seul coefficient non nul par ligne et par colonne, valant 1. Leur nom vient du fait que l'on peut associer à chaque permutation σ de n éléments une telle matrice $Q_\sigma \in \quad ^{n^2}$ par :

$$(Q_\sigma)_{ij} = \begin{cases} 1 & \text{si } \sigma(i) = j \\ 0 & \text{sinon.} \end{cases}$$

Par exemple, si $\sigma = \begin{pmatrix} 1 & 2 & 3 \\ 3 & 1 & 2 \end{pmatrix}$, alors $Q_\sigma = \begin{pmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}$.

Les matrices doublement stochastiques ont été rendues célèbres par le théorème de BIRKHOFF-VON NEUMANN, caractérisant le polytope B_n associé à l'ensemble des couplages parfaits d'un graphe biparti complet comme étant le polytope de toutes les matrices doublement stochastiques[137]. B_n s'appelle le **polytope d'affectation** d'ordre n. Voici quelques unes de ces propriétés : Défini dans $\quad ^{n^2}$, le polytope d'affectation B_n possède $n!$ sommets correspondant aux matrices de permutation, et est de dimension $(n - 1)^2$ ($\dim(B_n) = (n - 1)^2$). Le polytope P_n obtenu en projetant[138] B_n sur $\quad ^n$ par le vecteur $(1, 2, \ldots, n)$ est connu sous le nom de **permutoèdre** d'ordre n. Comme le polytope d'affectation B_n, le permutoèdre P_n possède $n!$ sommets correspondant aux permutations du vecteur $(1, 2, \ldots, n)$. Par contre, il est défini dans $\quad ^n$ et $\dim(P_n) = n - 1$. La Figure 30 nous présente deux exemples de permutoèdres. Pour de plus amples informations sur ces polytopes voir ZIEGLER [1995] ou BILLERA & SARANGARAJAN [1996].



(*a*) (*b*)

Figure 30 : Permutoèdres (*a*) d'ordre 3 : P_3 et (*b*) d'ordre 4 : P_4.

[137] Nous y reviendrons plus en détail au § 7.4.5.1.

[138] La projection d'un polytope P de $\quad ^{n \times n}$ sur $\quad ^n$ par un vecteur ligne x de $\quad ^n$ est le polytope de $\quad ^n$ obtenu par l'image de P par la projection $A \mapsto xA$ de $\quad ^{n \times n}$ dans $\quad ^n$.

Le polytope $P(x)$ obtenu en projetant B_n sur ^{n} par un vecteur quelconque x de dimension n est connu sous le nom de **multipermutoèdre**. Une description des facettes (en termes d'inégalités) est donné dans BILLERA & SARANGARAJAN [1996], ainsi que plusieurs propriétés combinatoires. Remarquons que cette description en système d'inégalités linéaires comporte généralement un nombre exponentiel d'inégalités en n. Nous en reparlerons au § 7.2.2.
L'ensemble des permutations d'ordre n étant en bijection avec l'ensemble des ordres totaux possibles sur n éléments[139], le nom de « permutoèdre » a souvent été utilisé pour désigner un autre polytope de $^{n^2}$: que nous appelons le **polytope des ordres totaux** d'ordre n. Les sommets de ce polytope sont les matrices $Q_{\succcurlyeq}$ obtenues à partir des ordres totaux $\succcurlyeq$ comme suit :

$$(Q_{\succcurlyeq})_{ij} = \begin{cases} 1 & \text{si } i \succcurlyeq j \\ 0 & \text{sinon.} \end{cases} \tag{6-31}$$

En reprenant l'exemple de l'ordre total : $3 \succcurlyeq 1 \succcurlyeq 2$, correspondant à la permutation de l'exemple précédent $\sigma = \begin{pmatrix} 1 & 2 & 3 \\ 3 & 1 & 2 \end{pmatrix}$, nous obtenons $Q_{\succcurlyeq} = \begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 0 \\ 1 & 1 & 1 \end{pmatrix}$.

Pour une description des facettes du polytope des ordres totaux, et des études polyédrales sur les polytopes associés aux ordres partiels et ordres d'intervalles via la relation (6-31), voir SCHULZ [1996].

Une fonction de la variable réelle F est **strictement Schur-concave**[140] (BERGE [1966], ROBERTS & VARBERG [1973]) si et seulement si pour toute matrice doublement stochastique Q, $F(Qx) > F(x)$, quand Qx est différent de x ou d'une permutation de x, et ce quelque soit le vecteur x de ^{n}.

6.2.3.3. Caractérisations de la Dominance de Lorenz

Le théorème qui suit, regroupe différentes caractérisations du principe de Lorenz $\succ_L$:

Théorème 2 (de Hardy, Littlewood et Polyà[141]) : Considérons deux vecteurs réels x, y de dimension n, dont les composantes sont rangées par ordre croissant : $x_1 \leq \ldots \leq x_n$, et $y_1 \leq \ldots \leq y_n$. Alors les conditions suivantes sont équivalentes :

(1) $\sum_{i=1}^{n} x(i) = \sum_{i=1}^{n} y(i)$, et pour tout $k \leq n$, $\sum_{i=1}^{k} x(i) \leq \sum_{i=1}^{k} y(i)$, avec l'inégalité stricte pour au moins un indice $k < n$.

(2) y peut être obtenu à partir de x, en effectuant une séquence finie non vide d'opérations de la forme :

$$\begin{aligned} &x^{\alpha+1}(i) = x^{\alpha}(i) + \varepsilon^{\alpha} \leq x^{\alpha}(j), \text{ pour tout } i < j \\ &x^{\alpha+1}(j) = x^{\alpha}(j) - \varepsilon^{\alpha} \geq x^{\alpha}(i),\ \varepsilon^{\alpha} > 0, \\ &x^{\alpha+1}(k) = x^{\alpha}(k), \qquad \text{si } k \neq i, j. \end{aligned} \tag{6-32}$$

et pour un au moins un α cette opération est un transfert de Pigou-Dalton.

(3) Si y est différent de x et n'est pas une permutation de x, alors il existe une matrice doublement stochastique Q, différente d'une matrice de permutation, tel que :

---

[139] Une permutation σ des n éléments d'un ensemble E peut être perçue comme un ordre total $\succcurlyeq_\sigma$ sur ces n éléments, en utilisant la relation : $i \succcurlyeq_\sigma j \Leftrightarrow \sigma^{-1}(i) \leq \sigma^{-1}(j)$, pour tout $i, j \in E$.
[140] Le nom « strictement S-concave » est aussi trouvé dans la littérature.
[141] Voir HARDY G., LITTLEWOOD J. & POLYÀ G. [1934] – *Inequalities* – Cambridge University Press, Londres. Ce théorème a été généralisé par DASGUPTA *et al.* [1973]. C'est cette version qui est présentée ici.

$$y = Qx$$

(4) Pour toute fonction F de la variable réelle définie sur ^{n} strictement Schur-concave sur $S(I_{PR})$,

$$\sum_{i=1}^{n} F(x(i)) < \sum_{i=1}^{n} F(y(i)) \qquad (6\text{-}33)$$

Pour une démonstration, voir par exemple DASGUPTA *et al.* [1973] ou SEN [1973].

La séquence finie d'opérations de la propriété (2) est une **généralisation du principe de Transfert de Pigou-Dalton**[142]. Ainsi, dans le contexte du problème de pure répartition, ces opérations s'interprètent par des transferts d'un individu mieux loti, vers un moins bien loti. Dès que nous pouvons effectuer des transferts de ressources d'un plus riche vers un plus pauvre pour passer d'une solution x vers une autre y, alors y est jugé moins inégal que x.

La Figure 31(*a*) montre le découpage obtenu de l'ensemble des solutions réalisables effectué par la dominance de Lorenz, sur l'instance du problème de pure répartition du début, à partir de la distribution de référence $x = (6, 2, 4)$. L'hexagone irrégulier (le multipermutoèdre $P(x)$) central représente l'ensemble des distributions qui Lorenz-dominent x. Les sommets de l'hexagone représentent les distributions équivalentes à x. Les ensembles en forme de V composant $B(x)$ représentent toutes les distributions qui sont Lorenz-dominés par x. Les 3 trapézoïdes et les 3 triangles contiennent toutes les distributions incomparables à x, c'est-à-dire les solutions réalisables où la courbe de Lorenz associée croise au moins une fois la courbe de Lorenz de x.

La propriété (3) du Théorème 2 permet d'identifier l'ensemble des solutions de meilleure qualité que x au multipermutoèdre $P(x)$ privé de ses sommets : les matrices de permutation.

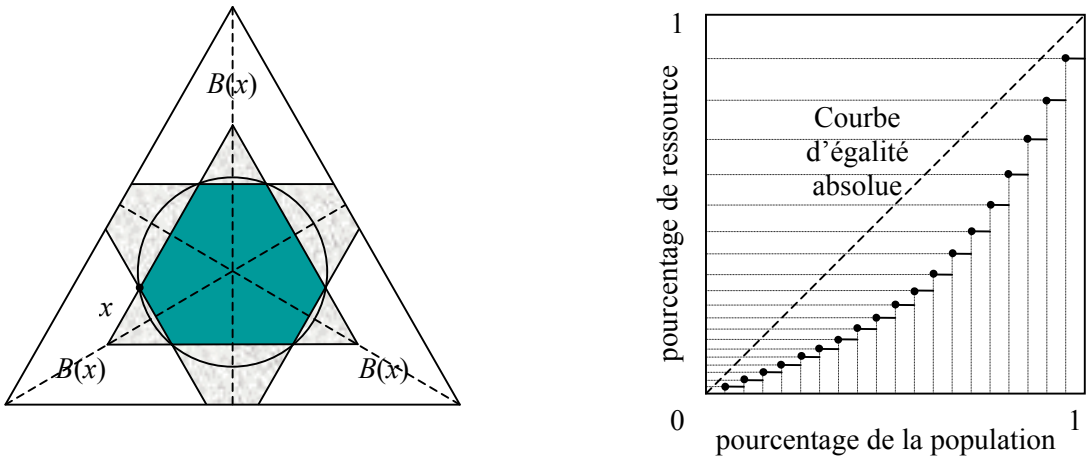


Figure 31 : (*a*) Dominance de Lorenz et Triangle de Kolm (*b*) Illustration d'une Courbe de Lorenz Discrète.

La propriété (4) nous informe que tout indicateur – donc vrai-critère – strictement Schur-concave est une extension de la dominance de Lorenz. Cette règle est l'intersection de tous les indicateurs strictement Schur-concaves. Prenons l'exemple des indicateurs statistiques d'inégalité : le coefficient de variation, la mesure d'entropie de Theil et le coefficient de Gini. Leur inverse respectifs sont des extensions de la dominance de Lorenz sur $S(I_{PR})$. Il en est autrement de l'inverse de la déviation standard des logarithmes aussi utilisé comme indicateur d'inégalité (voir DASGUPTA *et al.* [1973]). Remarquons que le coefficient de Gini d'une distribution peut être défini entièrement par la courbe de Lorenz : il correspond à l'aire délimitée

[142] Voir MARSHALL & OLKIN [1979] pour un historique.

par la ligne d'égalité absolue et la courbe de Lorenz de cette distribution (SEN [1973]). La déviation standard des logarithmes vérifie les axiomes d'égalité des proportions et d'invariance sous les permutations. Des travaux sont actuellement faits pour mieux identifier les cas de divergence entre cet indicateur – pourtant plutôt satisfaisant dans certains cas – et la dominance de Lorenz (voir par exemple FOSTER & OK [1999])

La dominance de Lorenz a été étudiée dans différentes disciplines, notamment :

– en physique : x Lorenz-domine y étant interprété par x décrit un état « plus chaotique » que y, en concevant $x(i)$ comme la probabilité d'un système de se trouver dans l'état i,
– en analyse matricielle et en algèbre linéaire : associée à la dominance de Pareto, la dominance de Lorenz permet d'effectuer des comparaisons de vecteurs de ^{n}. Ces travaux sont regroupés sous le nom de **théorie de la majorisation**. Nous y reviendrons au § 6.2.8.1.

La section suivante poursuit avec d'autres familles de vrai-critères strictement Schur-concaves.

La méthodologie qui a été présentée appelle à plusieurs remarques préalables :

1. L'utilisation d'une évaluation cardinale n'est pas nécessaire à la définition d'un concept. Comme le fait remarquer SEN [1973, chap. 3], rien dans la définition de l'inégalité n'indique que c'est une notion cardinale[143] ;
2. Un concept complexe comme l'inégalité possède différentes facettes, qui peuvent pointer dans différentes directions, et quelques fois un préordre complet peut ne pas émerger ;
3. La dominance de Lorenz est caractérisée par un ensemble d'axiomes commun à un grand nombre de vrai-critères. En présence de plusieurs concepts partiellement antagonistes, son utilisation permet d'identifier directement les axiomes sous-jacents en conflit, pour aboutir à une meilleure gestion de ces conflits[144]. Tandis qu'au contraire, l'utilisation d'un indicateur, même partiellement axiomatisé, possède l'inconvénient d'augmenter le nombre de conflits, la plupart n'apportant aucune information axiomatique pertinente. Cet encombrement peut aller à l'encontre d'un modèle préférentiel clair et pertinent.

6.2.4. Quelques Extensions de la Dominance de Lorenz

La dominance de Lorenz, bien qu'elle possède des propriétés jugées très attrayantes, entraîne relativement souvent l'incomparabilité. Elle reste silencieuse (incomparabilité) dans bien des comparaisons, se traduisant graphiquement par le croisement des courbes de Lorenz associées aux solutions (voir par exemple Figure 29). Pour y palier, des axiomes complémentaires ont été étudiés dans la littérature sociale et philosophique, à partir des indicateurs usuels. Ces études ont aussi abouti à la caractérisation de deux nouvelles classes infinies d'indicateurs généralisant la mesure d'entropie de Theil et le coefficient de Gini, tout en restant une extension de la dominance de Lorenz.

Le concept qui a été le plus étudié est la **sensibilité au transfert**. Intuitivement, il suggère qu'un transfert de ressource entre deux individus mal lotis a un effet plus favorable à l'équité qu'un même transfert entre deux individus bien nantis. Comme FOSTER & SEN [1997, A.4.3] indiquent dans le contexte des revenus :« *After all, why should a transfer between two millionaires have the same (or a greater) effect than the same transfer at the lower end of the distribution ?* ». Ce concept intuitif peut être traduit analytiquement de différentes manières. Par exemple (FOSTER & SEN [1997, A.4.3]) :

---

[143] Remarquons qu'une difficulté induite par l'utilisation d'information ordinale, est sa communication. La restitution d'information complexe à représenter, peut être en grande partie surmontée par des outils graphiques adéquats (§ 3.2.2.4) : Conception d'interfaces intelligentes, pouvant être dynamiques pour exprimer des informations très complexes. Utilisation des quatre dimensions (abscisse, ordonnée, couleur et temps) que met à disposition un écran d'ordinateur, comme support informationnel. Dans le cadre de la dominance de Lorenz, la courbe de Lorenz joue un rôle prépondérant dans la représentation de cette information.

[144] Nous parlons de gestion et non de résolution de conflits, voir SEN [1996].

Étant donné x et y deux solutions Lorenz incomparables, dont les courbes de Lorenz se coupent exactement une seule fois ; Si la courbe de Lorenz de x est initialement supérieur à celle de y, et que le coefficient de variation de x est inférieur ou égal à celui de y, alors x est jugée plus équitable que y :

$$\Leftrightarrow \quad x, y \in S(I_{PR}), \quad \left\{ \begin{array}{l} x \,\|_{\mathrm{L}}\, y \quad \text{et} \\ \exists 1 \le i < j < n \ \text{ tels que :} \left| \begin{array}{l} L_x(i/n) > L_y(i/n) \\ L_x(j/n) < L_y(j/n) \\ L_x(k/n) \ge L_y(k/n) \ \ \forall\, 0 \le k < i \quad \text{et} \\ L_x(k/n) = L_y(k/n) \ \ \forall\, i < k < j \\ L_x(k/n) \le L_y(k/n) \ \ \forall\, j < k \le n \end{array} \right. \\ CV(x) \le CV(y) \end{array} \right\} \Rightarrow x \succ_{RA} y.$$

Par exemple, la mesure de transfert de Theil T, vérifie cet axiome. Dans le cas du triangle de Kolm, prenons l'exemple donné en Figure 31 (*a*). Ce nouvel axiome réduit l'ensemble des solutions incomparables par rapport à x, comme suit : Le cercle représente l'ensemble des solutions équivalentes à la solution x. Les solutions des trois trapézoïdes extérieures au cercle sont moins équitables que x, alors que celles qui sont intérieures restent incomparables. Inversement, les solutions des trois triangles intérieures au cercle sont préférées, alors que celles extérieures sont incomparables.

6.2.5. Raffinements : Dominance de Lorenz et Relation de Préférence Valuée

Le concept d'inégalité est souvent perçu comme une notion vague, imprécise. Par conséquent, au lieu d'évacuer cette imprécision en utilisant un indicateur, il est possible d'en tenir compte, en faisant appel à l'incomparabilité, et d'user de relations binaires de préférence satisfaisant un ensemble d'axiomes nous paraissant important (ex. dominance de Lorenz et ses extensions).

Cependant, la rigidité engendrée par les relations binaires de préférence, en donnant le choix uniquement entre un verdict clair (préférence, indifférence, aversion) et l'absence de verdict (incomparabilité), ne laisse aucune place à des jugements gradués. En ce sens, un ensemble de règles d'agrégation, aboutissant à des relations binaires de préférence valuée (§ 3.3.2.2), et dédiées à la mesure de l'inégalité, a fait son apparition (BASU [1987] et OK [1995] parmi d'autres).

Ainsi, une relation binaire de préférence valuée μ_α sur $S(I_{\mathrm{PR}})$ est un **préordre flou de Lorenz** (OK [1995]), si l'attitude $\alpha = \succcurlyeq$, et pour tout couple de solutions $x, y \in S$,

a) $\mu_\alpha(x, y) = 1$ si et seulement si $x \succcurlyeq_{\mathrm{L}} y$,
b) $\mu_\alpha(x, y) = 0$ si et seulement si $y \succ_{\mathrm{L}} x$,
c) $\mu_\alpha(x, y) \in \,]0, 1[$ sinon.

Prenons comme exemple de préordre flou de Lorenz[145] :

$$\mu_{\succcurlyeq}(x, y) = \begin{cases} 1 & \text{si } x \text{ est une permutation de } y \\ E(x,y)/(E(x,y)+E(y,x)) & \text{sinon} \end{cases}, \text{ avec } E(x, y) = \sum_{i=1}^{n} \max\{L_x(i/n) - L_y(i/n), 0\}$$

Les points a) et b) de la définition sont vérifiés étant donné que lorsque $x \succ_{\mathrm{L}} y$, alors $E(x, y) > E(y, x) = 0$, et lorsque $x \simeq_{\mathrm{L}} y$, alors x est une permutation de y. Le point c) est directement induit par le fait que lorsque deux courbes de Lorenz se croisent (les solutions sont Lorenz incomparables), alors $E(x, y)$ et $E(y, x)$ sont tous deux strictement positifs.

[145] BASU [1987] utilisa une version stricte de cette relation binaire de préférence valuée :

$$\mu_{\succ}(x, y) = E(x, y) / (E(x, y) + E(y, x)).$$

6.2.6. Relaxations d'Informations de Faisabilité

Le problème de pure répartition fournit un cadre très limité à l'utilisation de la dominance de Lorenz, de ces extensions et raffinements. Seuls les problèmes obtenus à partir du problème de pure répartition, en ajoutant des contraintes de faisabilité (contraintes linéaires ou non, contraintes d'intégrité), peuvent en faire un usage intéressant. En conséquence, plusieurs travaux ont porté sur la conception de règles d'agrégation basées sur la dominance de Lorenz, pour les problèmes dont certaines contraintes de faisabilité sont relaxées. Les trois axiomes du problème de pure répartition, conditionnant l'ensemble des solutions réalisables, sont :

- La même population,
- Une quantité totale de ressource fixe,
- Une seule et même ressource.

Cette section est dédiée aux travaux qui ont été effectués lorsque ces hypothèses ne sont plus vérifiées. Les deux premiers points sont étudiés tout de suite, alors que le dernier point fait l'objet du § 6.2.7. Dans toute cette sous-section, nous supposons que les individus des différentes populations concernées ne peuvent revendiquer de traitement spécial basé sur les caractéristiques personnelles.

6.2.6.1. La Taille de la Population Varie

- La Taille de la Population Varie d'une Distribution à l'Autre.

Les problèmes de comparaison inter-pays, entre les régions d'un même pays, et autre, ont poussé les économistes se préoccupant d'équité et de pauvreté, à proposer des règles d'agrégation s'inspirant de la dominance de Lorenz, pour comparer des distributions admettant une population différente par le nombre.

Considérons donc un problème d'allocation de ressources Π tel que, pour une instance I_Π donnée la population varie : l'ensemble des solutions réalisables est un sous-ensemble de l'ensemble des vecteurs positifs de dimension finie : $S(I_\Pi) \subseteq \bigcup_{0<n<+\infty} (\quad_{+})^n$; Notons alors $E(x)$ la population associée à la distribution x, et n_x sa taille ($\Leftrightarrow$ card$(E(x)) = n_x$).

SEN [1973] proposa un axiome rendant la courbe de Lorenz invariante lorsqu'une distribution x est répliquée arbitrairement pour obtenir une nouvelle distribution $y = (x, \ldots, x)$.

Axiome d'Invariance sous les Réplications de la Population (PRI) : Étant donnés deux distributions x et y, et un entier $r > 0$ tels que :

$$n_y = rn_x, \qquad \text{et} \qquad y(i) = y(2i) = \ldots = y(ri) = x(i), \text{ pour tout } 1 \le i \le n_x.$$

Alors, $\qquad g_{RA}(y) = g_{RA}(x) \quad (\text{resp. } x \simeq_{RA} y)$

Remarquons que cet axiome nécessite que la quantité totale de ressource varie avec la distribution, car seuls les pourcentages de ressources cumulées sont comparés. Une règle d'agrégation satisfaisant cet axiome, est reliée à la courbe de Lorenz comme suit :

Quelles que soient les deux distributions x et y,

$$L_x = L_y \quad (\Leftrightarrow\ L_x(\lambda) = L_y(\lambda), \text{ pour tout } \lambda \in [0, 1]) \Rightarrow \qquad g_{RA}(x) = g_{RA}(y) \quad (\text{resp. } x \simeq_{RA} y)$$

La courbe de Lorenz d'une solution x n'est plus discrète, mais continue. Elle est choisie plus ou moins arbitrairement parmi les courbes de Lorenz continues passant par $L_x(i / n_x)$ pour tout $i \in \{0, \ldots, \text{card}(E(x))\}$. Cette courbe est majorée par la courbe de Lorenz continue obtenue à partir de x, en interpolant linéairement les valeurs intermédiaires entre $L_x(i / n_x)$ et $L_x((i + 1) / n_x)$ pour tout $0 \le i \le n_x - 1$.

Pour un couple de solutions x et y, il suffit de comparer les courbes de Lorenz discrètes formées de ppcm(n_x, n_y) individus[146] obtenues après conversion via l'axiome PRI des courbes initiales.

6.2.6.2. La Quantité Totale de Ressources à Distribuer Varie

- La Quantité Totale de Ressources à Distribuer Varie d'une distribution à l'autre.

Il serait intéressant d'étendre la dominance de Lorenz aux problèmes de distribution d'une ressource dont la quantité totale est variable d'une solution à l'autre. A priori, aucun axiome connu ne véhicule des idées d'équité, d'inégalité. Par conséquent, généralement, les autres axiomes sont apportés par les autres concepts du problème comme l'efficacité (§ 6.2.8.1), ou tout simplement des axiomes pertinents pour les solutions pratiques qu'elles entraînent (§ 6.2.8.2 et 6.2.8.3).

6.2.7. Le Cas des Ressources Multidimensionnelles

Nous nous intéressons ici à la généralisation du problème de pure répartition, au cas d'une distribution de plusieurs ressources non compensatoires. Depuis les travaux de SEN[147], les économistes, se sont beaucoup intéressés à la conception d'indicateurs mesurant l'inégalité, à partir de ressources formées de plusieurs commodités (ou attributs) incomparables. Les raisons sont multiples. Par exemple, les indicateurs de bien-être usuels – permettant de comparer les inégalités entre deux pays, ou de définir un seuil de pauvreté – ne tiennent compte que du revenu individuel de la population. Un indicateur tenant compte aussi de la santé individuelle, ou encore du degré d'alphabétisation, etc., qui plus est basé sur des axiomes solides comme le principe de transfert de Pigou-Dalton, aurait été plus pertinent pour comprendre la pauvreté et mettre en place des politiques plus adéquates.

6.2.7.1. Le Problème de Pure Répartition Multidimensionnelle

La généralisation du problème de pure répartition, au cas de plusieurs ressources non compensatoires, est appelé **problème de pure répartition multidimensionnelle** (PRM). Sa définition est la suivante :

Une instance $I_{PRM} = (E, R, \{Q(r), \text{ pour tout } r \in R\})$ est formée d'un ensemble de ressources R en quantité totale finie $Q(r) \in$ pour chaque $r \in R$, à distribuer en totalité à une population E finie. Une solution x est un élément de $^{\text{card}(R) \times \text{card}(E)}$, avec $x(r, i) \in$ la quantité de ressource r affectée à l'individu i. L'ensemble des solutions réalisables $S(I_{PRM})$ du problème est alors défini comme suit :

$$S(I_{PRM}) = \left\{ x \in {}^{\text{card}(R) \times \text{card}(E)} \;\middle|\; \begin{array}{l} x(r,i) \geq 0 \text{ pour tout } (r,i) \in R \times E \text{ et} \\ \sum_{i \in E} x(r,i) = Q(r) \text{ pour tout } r \in R \end{array} \right\}$$

Le but est donc de caractériser les préférences collectives (indicateurs, préordres, etc.) informant exclusivement sur le bien-être d'une population homogène décrite par I_{PRM}.

L'ensemble des solutions réalisables $S(I_{PRM})$ est un polytope de dimension $\text{card}(R)\cdot(\text{card}(E) - 1)$. Il correspond au polytope des b-couplages parfaits généraux[148] du graphe biparti complet $K_{\text{card}(R), \text{card}(E)}$.

---

[146] Le ppcm de deux entiers naturels est le **plus petit commun multiple** de ces deux nombres (REINHARDT & SOEDER [1997]).

[147] SEN Amartya Kumar [1976] – Real National Income – *Review of Economic Studies* 43 (1) : 19-39. Voir aussi SEN [1987], [1992].

[148] Pour un complément d'information sur ces types de couplages généralisés voir GONDRAN & MINOUX [1995], EISENBRAND *et al.* [2002], DAHL [1999*b*] ou encore SCHRIJVER [2003].

6.2.7.2. Dominance de Lorenz Multidimentionnelle : Quelques Axiomes

Quels sont les axiomes qui, *a priori* caractériseraient bien les préférences collectives émettant des jugements perçus comme équitables dans ce contexte multidimensionnel ?

Les axiomes s'inspirent de ceux utilisés dans le cas d'une seule ressource.

- L'équivalent multidimensionnel de l'axiome d'égalité des proportions utilise la solution x^* distribuant équitablement chaque ressource :

$$x^*(r, i) = \mu(r) \qquad \text{pour tout } (r, i) \in R \times E, \qquad \text{avec } \mu(r) = \frac{Q(r)}{\text{card}(E)}$$

  Si tous les individus ont le même panier d'attributs alors c'est l'égalité complète.
- L'axiome d'invariance sur les permutations s'applique aux paniers d'attributs obtenus par chaque individu. L'équité ne dépend pas de l'identité des individus.
- Par contre un équivalent multidimensionnel du principe de transfert de Pigou-Dalton pleinement satisfaisant, est plus difficile à concevoir.

Attardons-nous sur ce dernier point. Voici un exemple – inspiré de TSUI [1999] – montrant quelques singularités du multidimensionnel.

Considérons les quatre solutions réalisables d'un problème de pure répartition bidimensionnelle entre trois individus :

$$x_1 = \begin{pmatrix} 2 & 5 & 5 \\ 1 & 3{,}5 & 3{,}5 \end{pmatrix}, x_2 = \begin{pmatrix} 2 & 4 & 6 \\ 1 & 2 & 5 \end{pmatrix}, x_3 = \begin{pmatrix} 2 & 4 & 6 \\ 1 & 5 & 2 \end{pmatrix} \quad \text{et} \quad x_4 = \begin{pmatrix} 2 & 4 & 6 \\ 1 & 4 & 3 \end{pmatrix}$$

Une ligne r correspond à la distribution entre les individus de la ressource r, et une colonne i aux quantités des différentes ressources reçues par l'individu i. Considérons la distribution x_2. Nous remarquons que le premier individu est plus désavantagé que le deuxième au vu des deux commodités. Ces deux premiers individus sont eux plus mal lotis que le troisième individu. Comparons maintenant les différentes distributions. La solution x_1 est obtenue à partir de la solution x_2 en effectuant un transfert d'une unité de la première commodité et de 1,5 unités de la seconde commodité, du troisième individu vers le second plus désavantagé, pour obtenir une distribution équitable entre ces deux individus. La solution x_3 est obtenue à partir de x_2 en permutant simplement les quantités de la seconde ressource attribuées à la seconde et à la troisième personne. La solution x_4 est obtenue à partir de x_3 en effectuant un transfert de la seconde ressource entre la seconde et la troisième personne, du mieux servi en cette ressource (la seconde) vers le moins bien servi (la troisième).

Intuitivement, nous avons le rangement partiel suivant :

- La première solution semble clairement plus équitable que la seconde car les deux derniers individus possèdent le même panier de ressources dans x_1, alors qu'ils sont différents dans x_2 ;
- la troisième solution est plus juste (équitable) que la seconde, car x_2 affecte à la troisième personne plus des deux ressources qu'à la seconde, alors que x_3 gomme cette Pareto-dominance du panier d'attributs de la troisième personne par rapport à la seconde ;
- la quatrième solution est incomparable à la seconde et à la troisième solution au vu des seules informations données (les attributs sont incomparables).

Ces idées sous-jacentes présentées de manière informelle via l'exemple précédent, ont conduit à la formulation de trois axiomes adéquats[149]. Le premier, l'axiome de Lorenz uniforme, est une généralisation possible de la préférence stricte de la dominance de Lorenz :

Axiome de Lorenz Uniforme (UL) : Étant données deux distributions multidimensionnelles x et y de $S(I_{PRM})$, s'il existe une matrice bistochastique Q différente d'une matrice de permutation telle que : $x = yQ$, alors $x \succ_{RA} y$ (resp. $g_{RA}(x) > g_{RA}(y)$).

[149] D'autres axiomes sont donnés dans la bibliographie : MAASOUMI [1986], TSUI [1999], KOSHEVOY [1997], [1999], LIST [1999], SAVAGLIO [2001], [2002*a*], [2002*b*] parmi d'autres.

Lorsque le nombre d'attributs card(R) = 1, l'axiome UL correspond à la préférence stricte de la dominance de Lorenz. Avec l'exemple précédent, UL indique que la solution x_1 est plus juste que la solution x_2, la matrice bistochastique Q étant la suivante :

$$Q = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0{,}5 & 0{,}5 \\ 0 & 0{,}5 & 0{,}5 \end{pmatrix}.$$ Plus généralement, nous avons : $x_1 = x_2Q$, $x_1 = x_3Q$ et $x_1 = x_4Q$.

Par contre cet axiome ne permet pas de ranger les solutions x_2 et x_3, car il ne se préoccupe que des lignes des matrices solutions. L'axiome qui permet de véhiculer cet aspect de l'équité est le transfert corrélation croissante (CIT). D'abord, considérons deux vecteurs x et y de ^{n} et notons min$\{x, y\}$ et max$\{x, y\}$ les vecteurs de ^{n} obtenus comme suit : min$\{x, y\}$ = (min$\{x(1), y(1)\}$, …, min$\{x(n), y(n)\}$), et max$\{x, y\}$ = (max$\{x(1), y(1)\}$, …, max$\{x(n), y(n)\}$). Alors l'axiome CIT se définit comme suit :

Axiome de Transfert Corrélation Croissante (CIT) : Étant données deux distributions multidimensionnelles x et y de $S(I_{PRM})$, s'il existe un couple d'indices de colonnes i et j, $i < j$, tel que :

$$y(\cdot, i) = \min\{x(\cdot, i), x(\cdot, j)\},$$
$$y(\cdot, j) = \max\{x(\cdot, i), x(\cdot, j)\}, \text{ et}$$
$$y(\cdot, k) = x(\cdot, k), \qquad \text{pour tout } k \neq i, j.$$

alors, $x \succcurlyeq_{RA} y$ (resp. $g_{RA}(x) \geq g_{RA}(y)$).

La version stricte est obtenue en supposant aussi que $x(\cdot, i) \neq y(\cdot, i)$.

Une version généralisée de cet axiome est obtenue comme suit :

Axiome de Lorenz Corrélation Croissante (CIL) : Étant données deux distributions multidimensionnelles x et y de $S(I_{PRM})$, si y peut être obtenu à partir de x par permutation des colonnes et par une séquence finie de transferts corrélation croissante, avec au moins une stricte, alors $x \succ_{RA} y$ (resp. $g_{RA}(x) > g_{RA}(y)$).

Remarquons qu'une idée d'agrégation qui vient souvent à l'esprit consiste à effectuer un ensemble de transferts de Pigou-Dalton (PDT) sur chaque ressource sans tenir compte des autres ressources[150]. Cette règle d'agrégation peut entraîner des jugements très contestables en termes d'équité, car, en se basant sur l'exemple précédent, elle juge les solutions x_2 et x_3 indifférentes, jugement qui va à l'encontre des axiomes CIT et CIL). Cet exemple illustre bien les erreurs entraînées par l'agrégation directe d'indicateurs mono-attributs sans soucis des influences croisées (par exemple la somme pondérée ou la règle de Pareto).

---

[150] Cette idée se traduit par :

La solution y est préférée à la solution x, si et seulement si :

- Pour tout $r \in R$, $y(r, \cdot) \succcurlyeq_L x(r, \cdot)^T$, et $\exists\, r \in R$, $y(r, \cdot)^T \succ_L x(r, \cdot)^T$; avec $z(r, \cdot)$ le vecteur ligne d'indice r de la solution $z \in$ $^{\text{card}(R) \times \text{card}(E)}$, et $z(r, \cdot)^T$ la transposée de $z(r, \cdot)$.

Cette règle se traduit sous forme matricielle par :

- Il existe card(R) matrices bistochastiques de dimension card(E) × card(E), telles que :

$$y = \sum_{r \in R} \delta_{r,r} \cdot x \cdot Q_r$$

et il existe au moins une ressource r telle que Q_r n'est pas une matrice de permutation ; avec $\delta_{k,l}$ la matrice carrée de card(R) lignes et colonnes, connue sous le nom de symbole de Kronecker :

$$\delta_{k,l} = \begin{cases} 1 & \text{pour } k = l \\ 0 & \text{pour } k \neq l \end{cases}.$$

En d'autres mots, cette règle est équivalente à établir une hiérarchie agrégative en agrégeant la distribution des ressources de chaque attribut avec la règle de Lorenz, puis en agrégeant ces règles via la règle de Pareto.

Dans son article [1999], TSUI caractérise un ensemble infini d'indicateurs d'inégalités dont leur inverse satisfait entre autre UL et CIL. Ces indicateurs permettent d'effectuer des investigations empiriques usant de mesures d'inégalité multidimensionnelle.
De par sa complexité, la répartition multidimensionnelle a donné lieu à moins de travaux que sa version unidimensionnelle. Récemment, DAHL [1999*a*], [1999*b*] a mené des études polyhédrales dans le cadre multidimensionnel. Ainsi, par rapport au cas unidimensionnel (§ 6.2.3.2), le polytope d'affectation est remplacé par le polytope des matrices ligne-stochastiques[151], et le multipermutoèdre par le markotope[152].
D'autres travaux ont porté sur la représentation graphique de cette information qualitative (voir note en bas de page numéro 143), notamment avec le **zonotope de Lorenz**, qui est une généralisation bidimensionnelle de la courbe de Lorenz (KOSHEVOY [1997], DAHL [1999*a*], [1999*b*]).

Lorsque la population varie, une version multidimensionnelle de l'axiome d'invariance sous les réplications de la population, notée MPRI par la suite, a été définie dans les différents articles précités.

6.2.8. Dominance de Lorenz et Autres Critères

Des études ont aussi été menées sur des extensions de la dominance de Lorenz, satisfaisant des axiomes complémentaires ne traduisant pas nécessairement un jugement d'équité. Les axiomes supplémentaires sont rattachés à d'autres principes comme :

- l'efficacité (§ 6.2.8.1) se préoccupant d'accroître la masse globale des avantages individuels sans s'occuper de leur répartition.
- la **praticité** (§ 6.2.8.2 et 6.2.8.3) s'intéressant à identifier des préférences collectives pouvant avantageusement être utilisées dans des processus de décision spécifiques (protocole de résolution, usage, décision à prendre).

Nous présentons ci-dessous les extensions obtenues en ajoutant l'un des axiomes suivants : l'efficacité de Pareto, l'invariance face aux changements d'échelle, la décomposabilité additive et la consistance de sous-groupe.

6.2.8.1. Efficacité de Pareto : La Majorisation

De même que l'équité, l'efficacité préférentielle est l'un des principes les plus fréquemment cités comme brique de base pour une agrégation. L'efficacité est caractérisée par l'axiome de Pareto (§ 4.2.2.2) lequel requiert la maximisation de la quantité de ressources affectée à chaque individu. La dominance de Pareto intervient lorsque la quantité totale Q de ressources n'est plus fixée, dans le cas contraire, les solutions réalisables sont toutes incomparables (§ 6.2.2.2).

Considérons donc un problème Π d'allocation d'une ressource telle que, pour une instance I_Π donnée :

- la population E est connue explicitement ($n = \mathrm{card}(E)$), les individus n'étant pas socialement différentiables (§ 6.2.1) ;
- l'ensemble des solutions réalisables est un sous-ensemble de l'ensemble des vecteurs positifs de dimension n : $S(I_\Pi) \subseteq {}^{n}_{+}$; avec $Q(x)$ la quantité totale de ressources de la distribution x, et $\mu(x)$ la moyenne associée.

---

[151] Une **matrice ligne-stochastique** (ou **matrice de Markov**) de dimension $m \times n$ est une matrice à coefficients positifs ; dont chacune de ses lignes est formée de coefficients dont la somme vaut 1.
[152] Le **markotope** $P(x, k)$ d'une matrice x de ${}^{m \times n}$ et d'un entier positif k, est un polytope de ${}^{m \times k}$, dont les éléments sont obtenus en effectuant la multiplication matricielle de x par toutes les matrices ligne-stochastiques de dimension $n \times k$.

Intéressons-nous aux préférences collectives vérifiant les axiomes sous-jacents à la dominance de Lorenz (invariance sous les permutations (IP) et principe de transfert de Pigou-Dalton (PDT)), ainsi que l'axiome de Pareto.

Tout d'abord, ce système axiomatique est consistant. En effet, le système axiomatique sous-jacent à la dominance de Lorenz s'applique à des couples de solutions où les individus sont en désaccord, alors que l'axiome de Pareto s'applique à des couples où les individus ne sont justement pas en conflit.

Notons $\mathcal{F}$ le profil simple formé des vrai-critères $g_i = (f_i(x) = x(i), \geq)$ pour tout $i \in E$.
Le système axiomatique formé de la fermeture transitive de l'axiome d'invariance sous les permutations et de l'axiome de Pareto est appelé système axiomatique Pareto symétrique. La règle d'agrégation associée est appelée dominance Pareto symétrique.

Système axiomatique : PS($\mathcal{F}$) = Fermeture transitive (Pareto ($\mathcal{F}$) $\cup$ IP($\mathcal{F}$)).

La règle d'agrégation, associée au système axiomatique formé de la fermeture transitive du système axiomatique de Lorenz et de l'axiome de Pareto, est connue sous le nom de dominance de Lorenz généralisée ou **GL-dominance** ou encore **majorisation**.

Système axiomatique : GL($\mathcal{F}$) = Fermeture transitive (Pareto($\mathcal{F}$) $\cup$ IP($\mathcal{F}$) $\cup$ PDT($\mathcal{F}$)).

La Figure 32 (*a*) (resp. Figure 32 (*b*)) montre la découpe de l'ensemble des solutions ^{2} effectuée par la dominance Pareto symétrique (resp. la GL-dominance) en un point x. La partie inférieure sombre correspond aux solutions dominées par x. La partie intermédiaire en gris-clair forme l'ensemble des solutions incomparables à x. Le symétrique de x est la seule solution qui lui soit indifférente, et la partie supérieure correspond à l'ensemble des solutions dominant x.

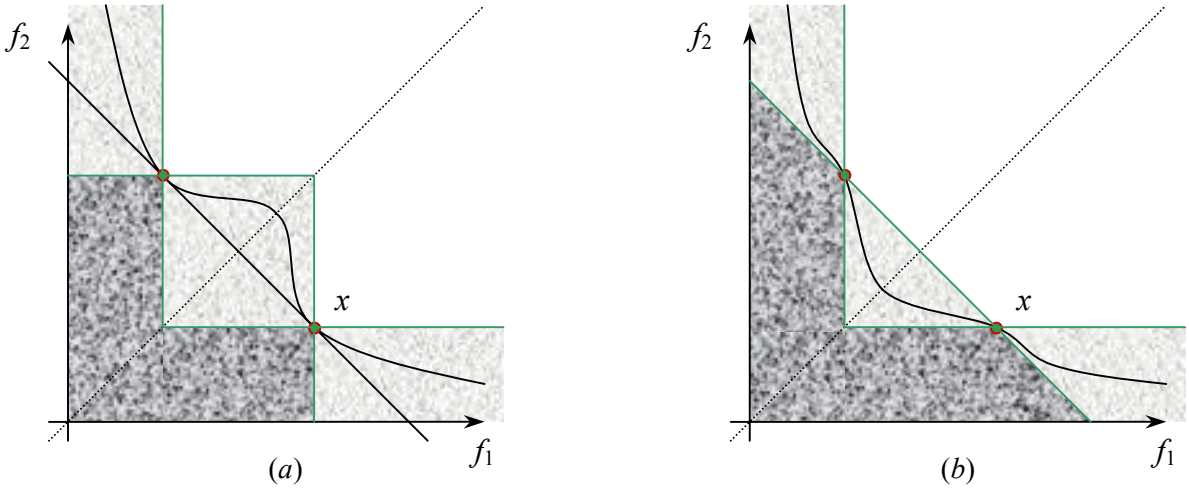


Figure 32 : Découpe de l'espace des critères ^{2} effectuée par (*a*) la dominance Pareto symétrique, et (*b*) la GL-dominance.

L'ensemble des vrai-critères satisfaisant l'axiomatique Pareto symétrique est l'ensemble des fonctions continues symétriques sur ^{n}. Deux exemples de vrai-critères symétriques sont donnés à la Figure 32 (*a*). La seule fonction linéaire symétrique est la somme simple (les poids sont non nuls et tous égaux).
La GL-dominance $\succcurlyeq_{GL}$ est reliée à la courbe de Lorenz comme suit :
Étant donné $x, y \in S(I_\Pi)$, $\quad x \succcurlyeq_{GL} y \Leftrightarrow \mu(x)\cdot L_x(i / n) > \mu(y)\cdot L_y(i / n)$ pour tout $0 < i \leq n$.
Les vrai-critères dont le préordre sous-jacent est une extension de la GL-dominance sont exactement les fonctions Schur-concaves[153] sur ^{n}.

[153] L'inégalité stricte est perdue en passant de la Lorenz-dominance à la GL-dominance.

Remarquons que la mesure d'entropie de Theil $T(\cdot)$, le coefficient de variation $CV(\cdot)$ et le coefficient de Gini $G(\cdot)$ sont strictement Schur-convexes sur $S(I_{PR})$, pour toute instance I_{PR} de problème de pure répartition sur n individus. Mais ces indicateurs ne sont pas Schur-convexes sur ^{n}. Ils rentrent en conflit avec l'axiome de Pareto. Par contre l'indicateur $\mu(\cdot)\cdot(1 - f_{\text{stat}}(\cdot))$, $f_{\text{stat}}(\cdot)$ étant l'un des trois indicateurs précédents, est Schur-concave sur ^{n}.

Plus généralement, un théorème équivalent au Théorème 2 (appelé aussi théorème de Hardy, Littlewood et Polyà) peut être formulée. Nous ne le reprenons pas ici et renvoyons par exemple à MARSHALL & OLKIN [1979].

L'axiome de Pareto a aussi été utilisé dans le cas multidimensionnel ainsi que dans le cas d'une population variable (axiomes PRI et MPRI). Le livre de MARSHALL & OLKIN [1979] est une présentation exhaustive, détaillée et entièrement consacrée à la majorisation. Voir aussi FOSTER & SEN [1997].

6.2.8.2. Invariance Face aux Changements d'Échelle

Alternativement à l'axiome de Pareto, pour traiter les problèmes d'allocation de ressources dont la quantité totale varie d'une solution à l'autre, un autre axiome – très utilisé en économie pratique – stipule l'invariance par homothétie. Sa définition est la suivante :

Axiome d'Invariance par Changement d'Échelle (RS) : Étant données deux distributions x et y, et un réel $\lambda > 0$ tels que : $y(i) = \lambda \cdot x(i)$, pour tout $1 \leq i \leq n$.
Alors, $g_{RA}(y) = g_{RA}(x)$ (resp. $x \simeq_{RA} y$)

L'intérêt porté à cet axiome est généralement d'ordre pratique car, associé par fermeture transitive à l'axiomatique sous-jacente à la dominance de Lorenz, il correspond exactement à la comparaison des courbes de Lorenz :
Étant donné $x, y \in S(I_{\Pi})$, $L_x(i / n) \geq L_y(i / n)$ pour tout $i \in E \Rightarrow x \succcurlyeq_{RA} y$ (resp. $g_{RA}(x) = g_{RA}(y)$). Il en est de même de l'indifférence et de la préférence stricte.

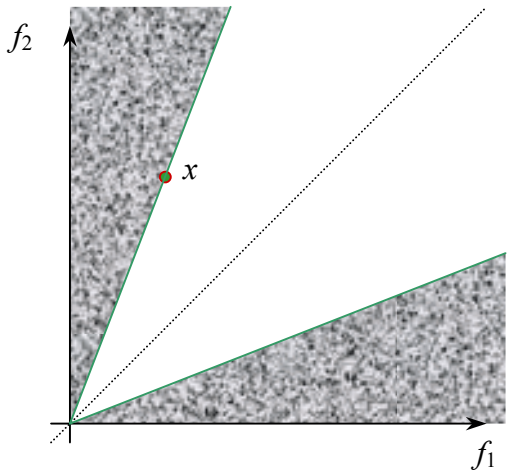


Figure 33 : Découpe de l'espace des critères ^{2} engendré par la dominance de Lorenz invariante aux changements d'échelle.

Le préordre engendré est total dans le cas d'une population de deux individus. La Figure 33 illustre ce cas en une solution x. Les solutions préférées à x forment l'intérieur du cône central. Les solutions dominées par x sont les deux triangles de couleur sombre, et les solutions indifférentes à x forment la frontière du cône auquel l'origine est soustraite.
Dans le cas général, le préordre n'est que partiel.

Notons $\tau(L \vee RS)$ le système axiomatique formé de la fermeture transitive (τ) de l'axiomatique sous-jacente à la dominance de Lorenz (L) et de l'axiome d'invariance par changement d'échelle (RS). L'axiome de Pareto et l'axiome RS peuvent entrer en conflit, étant donné qu'un rapport d'homothétie supérieur à 1 correspond à une indifférence pour RS, et correspond à une amélioration pour tous les individus. La comparaison de la Figure 32 et Figure 33, illustre bien ce conflit : Les solutions maximales pour la majorisation ne sont pas

forcément celles de $\tau(L \vee RS)$ pour une même solution. Rappelons que ce conflit est valable dans le cas général. Par contre, en terme de choix collectif, la forme de l'ensemble des solutions réalisables $S(I_\Pi)$ peut être telle qu'il existe des solutions qui soient choisies par l'intersection de ces deux axiomes.

L'inverse d'un indicateur satisfaisant les axiomes L, RS et PRI, ou leurs variantes multidimensionnelles, est appelé **mesure relative d'inégalité**. Dans le cas d'une ressource unidimensionnelle, la mesure d'Entropie de Theil et les coefficients de Gini et de Variation sont des mesures relatives d'inégalité.

6.2.8.3. Décomposabilité Additive et Consistance de Sous-Groupe

Les mesures relatives d'inégalité sont une très belle illustration d'indicateurs combinant des axiomes d'usage (RS), et des axiomes de définition du problème (axiomes de Lorenz). Deux autres exemples d'axiomes d'usage sont la décomposabilité additive et la consistance de sous-groupes.

Ces deux axiomes considèrent un problème de pure répartition où la population E est caractérisée par un ensemble de propriétés pertinentes (âge, sexe, race, etc.) répartissant (partition) celle-ci en groupes homogènes $\mathcal{G}$. Notons x_k la restriction de la solution x au groupe d'individus $G_k \in \mathcal{G}$, et $\bar{x}$ la solution effectuant une moyenne des composantes de x par groupe :

$$\bar{x}(i) = \frac{1}{\text{card}(G_k)} \sum_{j \in G_k} x(j) \quad \text{pour tout } i \in G_k, \text{ et pour tout } G_k \in \mathcal{G}.$$

Nous notons dans cette section indifféremment g_{RA} (resp. $\succcurlyeq_{RA}$) pour désigner les préférences collectives sur $S(I_{PR})$ avec $I_{PR} = (E, Q)$, ou pour désigner tout sous-problème de pure répartition de I_{PR}, c'est-à-dire tout problème de pure répartition d'instance (E', Q') avec $E' \subseteq E$ et $Q' \leq Q$. Le contexte permet de désigner dans quel espace les préférences sont appliquées.

❖ Décomposabilité Additive

L'axiome de décomposabilité additive ne s'applique que sur les indicateurs et se formule comme suit :

Axiome de Décomposabilité Additive (DA) :

$$\text{Pour tout } x = (x_1, \ldots, x_{\text{card}(\mathcal{G})}) \in S(I_{PR}) \qquad g_{RA}(x) = g_{RA}(\bar{x}) + \sum_{k=1}^{\text{card}(\mathcal{G})} w(x_k) \cdot g_{RA}(x_k)$$

avec $w(x_k)$ un poids associé au groupe G_k dépendant généralement du pourcentage d'individus formant le groupe, et de la quantité moyenne de ressource de G_k.

Cet axiome permet de décomposer l'indicateur en des termes inter- et intra-groupe. Le terme inter-groupe $g_S(\bar{x})$ informe sur la valeur de l'indicateur si chaque groupe est remplacé par la quantité moyenne du groupe, permettant ainsi de connaître l'apport des variations entre groupes. Le terme intra-groupe (la somme pondérée) est la moyenne pondérée de la valeur de l'indicateur pour chaque groupe, informant ainsi de l'apport de chaque groupe à l'indicateur global.

Par exemple, le coefficient de variation au carré et la mesure Theil sont additivement décomposables, contrairement au coefficient de Gini. Pour un complément, voir BOURGUIGNON [1979] et FOSTER & SEN [1997]. Dans le cas de plusieurs ressources voir MAASOUMI [1986] et TSUI [1999].

❖ Consistance de Sous-Groupe

L'axiome de consistance de sous-groupe requiert que si l'équité augmente pour un groupe lorsque l'on passe de la solution x à la solution y, l'état des autres groupes restant équivalent, alors x est préférée à y :

Axiome de Consistance de Sous-Groupe (SGC) : Étant données deux solutions $x = (x_1, \ldots, x_{\mathrm{card}(\mathcal{G})})$ et $y = (y_1, \ldots, y_{\mathrm{card}(\mathcal{G})})$ de $S(I_{PR})$, et un indice $k \in \{1, \ldots, \mathrm{card}(\mathcal{G})\}$;

$$[x_k \succ_{RA} y_k] \text{ et } [x_l \simeq_{RA} y_l \text{ pour tout } l \neq k] \Rightarrow x \succ_{RA} y$$

Ce type d'axiome est intéressant pour des questions pragmatiques comme : Comment assurer la cohérence entre les politiques régionales et nationales en matière de réduction des inégalités (FOSTER & SEN [1997]). Un indicateur d'inégalité satisfaisant l'axiome SGC satisfait aussi DA. D'autres résultats et références sont disponibles dans FOSTER & SEN [1997].

Pour conclure, les systèmes d'axiomes non conflictuels, peuvent servir de base à la conception de relations de préférences valuées pour graduer les préférences, comme cela a été effectué pour la dominance de Lorenz au § 6.2.4. Une « fuzzification » peut aussi être utilisée lors de conflits dissymétriques : parmi les axiomes en conflit, certains sont plus importants que d'autres. Malgré le conflit, pour indiquer une préférence faible en faveur des axiomes plus importants, une règle d'agrégation floue peut être utilisée.

Une définition plus juste du problème de pure répartition serait d'expliciter l'information interpersonnelle liant très fortement les critères. Nous en discutons à la prochaine section.

6.3. Équité, Efficacité et Information Préférentielle Étendue Ordinale

Jusqu'à maintenant, les règles d'agrégation d'équité présentées supposent une population homogène en référence à des ressources comportant un ou plusieurs attributs.

Nous allons maintenant présenter les travaux traitant de l'équité dans un cadre moins exigeant en information préférentielle. Tout d'abord, nous éclaircissons les hypothèses préférentielles individuelles et interpersonnelles sous-jacentes à la dominance de Lorenz et ses extensions (§ 6.3.1). Ensuite, du § 6.3.2 au § 6.3.4, l'équité et l'efficacité sont présentées dans le cadre des préférences individuelles décrites par une unique relation binaire de préférence étendue (§ 4.1.3). La dernière sous-section (§ 6.3.5) est consacrée à l'étude des possibilités d'utilisation de la dominance de Lorenz dans un cadre aussi général, où l'être humain est d'abord caractérisé par sa diversité.

6.3.1. Dominance de Lorenz et Comparaisons Interpersonnelles de Type Sympathie Étendue

Cherchons à définir *a posteriori* les hypothèses préférentielles sous-jacentes aux dominances présentées précédemment.

Réécrivons le problème de pure répartition en utilisant les profils étendus vus au § 4.1.3.1. Les hypothèses de départ (par exemple : personne ne peut se prévaloir de ses caractéristiques personnelles) informent aussi sur les préférences interpersonnelles : les quantités de biens sont comparables d'un individu à un autre et d'une solution à une autre. En d'autres termes, quels que soient les individus i et j de E et pour toutes distributions x et y de $S(I_{PR})$, si $x(i) \geq x(j)$ alors la position de i dans l'état x est au moins aussi bonne que la position de j dans ce même état. En termes de comparabilité interpersonnelle, la quantité de ressource $x(i)$ reçue par l'individu i lorsque la solution x est adoptée, est interprétée comme la valeur $u(x, i)$ attribuée à l'individu i dans l'état social x. En d'autres mots, le modèle de préférence du problème de pure répartition est formé d'un seul vrai-critère étendu $g = (u, \geq)$ sur $S(I_{PR}) \times E$, avec $u(x, i) =$

$x(i)$, émis par le spécificateur du problème, ou le planificateur, etc.. Celui-ci s'appuie directement sur les quantités de ressources pour formuler ses préférences.
Les vrai-critères étendus sont formellement équivalents à $\mathrm{card}(E)$ vrai-critères simples, dont les valeurs sont de même dimension. Par exemple, si les ressources sont des volumes, des masses, de la monnaie ou de l'utilité, les quantités doivent s'exprimer dans la même unité : volumique, massique, monétaire ou de satisfaction.

Dans le cas du problème de pure répartition multidimensionnelle, la quantité de ressource $x(r, i)$ est interprétée comme l'avis $u_r(x, i) \in \mathcal{U}(S(I_{PRM}) \times E, \quad)$ de l'individu émetteur r sur la position de l'individu observé i dans l'état social x. Le modèle préférentiel est alors un profil de type sympathie étendue :

$$\mathcal{F}(R, E, S(I_{PRM})) = \{g_r = (u_r, \geq), \text{ pour tout } r \in R\} = \mathcal{F}(R, S(I_{PRM}) \times E).$$

Toute préférence collective satisfaisant le principe de transfert de Pigou-Dalton, est au moins niveau comparable (§ 4.1.3.4). Si on ne peut pas faire de telles comparaisons, cet axiome n'est plus applicable (SEN [1973, p. 43-46]). Qu'en est-il de la dominance de Lorenz et de ses extensions ? Elles utilisent un axiome de comparabilité plus élevé, car elles admettent que l'on puisse sommer les préférences individuelles au même titre que les quantités de ressources qui les supportent (MANDLER [2001]).

Nous avons jusqu'ici supposé que l'évaluation de notre population est homogène par rapport à des ressources uni- ou multidimensionnelles. Ce cadre peut être jugé trop restrictif. La dominance de Lorenz généralisée est applicable à n'importe quel vrai-critère étendu $g = (u, \geq)$ sur $\quad^{\mathrm{card}(E)} \times E$, ne vérifiant pas forcément $u(x, i) = x(i)$. Il suffit qu'il existe une fonction f telle que $u(x, i) = f(x(i))$. Par conséquent, si les préférences individuelles d'un individu se modélisent par un vrai-critère étendu sur $\quad^{\mathrm{card}(E)} \times E$, alors la dominance de Lorenz généralisée peut être utilisée. Il en est de même pour la dominance de Lorenz généralisée multidimensionnelle (MGL). La dominance de Lorenz généralisée peut aussi être utilisée dans le cas multidimensionnel. Les préférences individuelles se modélisent alors par un vrai-critère étendu sur $\quad^{\mathrm{card}(R) \cdot \mathrm{card}(E)} \times E$. Il en est de même de la dominance MGL, où un ensemble arbitraire d'individus formule des vrai-critères individuels étendus sur les positions (x, i). nous y reviendrons au § 6.3.5.

Mais, sorti de ce cadre qu'est il possible de faire ? L'incertitude fait partie intégrante de l'information préférentielle (§ 3.3). Est-il possible d'adapter la dominance de Lorenz et ses extensions à d'autres fonction-critères comme les critères d'intervalles (§ 3.3.3.3) ? Lorsque l'information préférentielle individuelle est encore plus pauvre et admet l'incomparabilité, est-il possible de formuler des jugements véhiculant des principes d'équité ?

Depuis les années 60, plusieurs travaux ont porté sur la conception de règles d'agrégation véhiculant des principes d'équité et d'efficacité. Ces règles d'agrégation ont la particularité de synthétiser une information préférentielle individuelle de type sympathie étendue ordinale. Plus précisément, les préférences individuelles sont des relations binaires de préférence étendues. Le reste de cette section les présente.

Ainsi, au § 6.3.2, le cas d'un profil étendu est présenté, suivit du cas des profils quelconques de relations binaires de préférence étendues (§ 6.3.3). L'essentiel des travaux porte sur la donnée d'un préordre complet étendu. Les différents résultats sont exposés dans un cadre moins structuré. Il sera précisé avec les éventuelles adaptations nécessaires. Au § 6.3.4, nous effectuons une ouverture sur les procédures interactives conçues en théorie des jeux pour résoudre des problèmes d'allocation de ressources. Et nous terminons (§ 6.3.5) par les systèmes combinant l'approche basée sur les préférences individuelles étendues, et celle basée sur les ressources.

6.3.2. Équité et Efficacité pour l'Agrégation de Relation Binaire de Préférence Étendue

Considérons donc une instance I_Π d'un problème Π, dont la description est la suivante :

S un ensemble des solutions réalisables donné en extension ou non,

I_e une population formée des acteurs sujets à comparaison, et

$\mathcal{F}$ un profil étendu sur $S \times I_e$ dont les préférences sous-jacentes de l'unique émetteur sont une relation binaire de préférence étendue notée $\succcurlyeq \in \mathcal{R}(\text{RBP}, S \times I_e)$. L'émetteur du profil est arbitraire, celui-ci pouvant donc être un individu extérieur à I_e, ou l'ensemble I_e si la population considérée est souveraine, etc.

Le but est de concevoir une règle d'agrégation, véhiculant des principes d'équité et d'efficacité, aboutissant à des préférences collectives structurées en relation binaire de préférence.

Dans un tel cadre, plusieurs règles d'agrégation et axiomes d'équité ont vu le jour. La règle d'équité de Suppes-Sen est présentée au § 6.3.2.1, suivie des règles du Maximin et Leximin (§ 6.3.2.2) très connues en RO. Nous clôturons cette sous-section par la présentation de plusieurs axiomes et règles d'agrégation traduisant une forme ordinale du principe de transfert de Pigou-Dalton (§ 6.3.2.3).

6.3.2.1. Équité de Suppes-Sen

La **règle d'équité de Suppes-Sen** SSE définit une relation binaire de préférence $\succcurlyeq_{\text{SSE}}$ sur S, à partir d'un préordre total étendu $\succcurlyeq$, comme suit : Considérons deux solutions x et y de S.

$$x \succ_{\text{SSE}} y \quad \Leftrightarrow \quad \text{il existe une permutation } \sigma \text{ des individus de } I_e \text{ telle que :} \begin{cases} \text{pour tout } i \in I_e, \quad (x,i) \succcurlyeq (y,\sigma(i)) \\ \text{il existe } j \in I_e \text{ vérifiant :} \quad (x,j) \succ (y,\sigma(j)) \end{cases}.$$

$$x \simeq_{\text{SSE}} y \quad \Leftrightarrow \quad \text{il existe une permutation } \sigma \text{ des individus de } I_e \text{ telle que :} \quad \text{pour tout } i \in I_e,\ (x, i) \simeq (y, \sigma(i)).$$

La partie asymétrique de $\succcurlyeq_{\text{SSE}}$ est un axiome appelé **principe de Suppes**[154]. Cet axiome signifie que : la solution x est plus équitable que la solution y, s'il existe une permutation des individus de I_e telle que l'émetteur arbitraire, juge que la position de chacun en x est au moins aussi bien que celle de la personne correspondante dans y ; et s'il existe un individu dont la position en x est préférée à la position de la personne correspondante dans y. La partie symétrique, l'**indifférence de Suppes**, juge deux solutions indifférentes s'il existe une permutation des individus de I_e qui, pour chaque individu fait correspondre un individu dont les positions sont jugées indifférente. Cet axiome est une version ordinale de l'axiome IP d'invariance sous les permutations (§ 6.2.1).

La règle d'équité de Suppes-Sen est une extension de la règle de Pareto (SEN [1970, chap. 9 et 9*]), et est souvent considérée comme la version interpersonnelle de la règle de Pareto. Par conséquent, elle fournit une relation respectant l'unanimité, tout en étant plus riche que la règle de Pareto. $\succcurlyeq_{\text{SSE}}$ forme un préordre partiel (conservation de la transitivité) sur S. La preuve est assez directe. Cette règle peut être étendue dans le cas de relation binaire de préférence étendue non nécessairement transitive et complète. Développons :

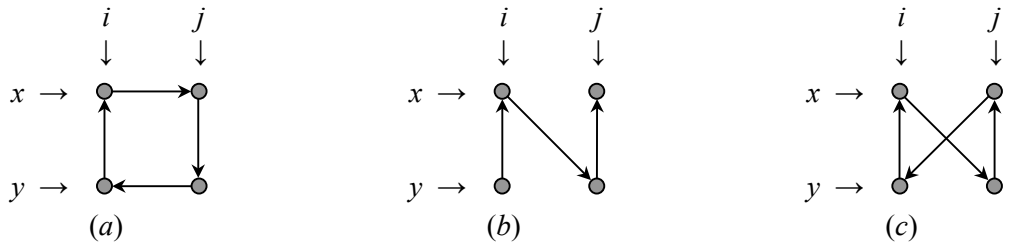
D'abord, voici une propriété des relations binaires de préférence étendues qui sera utile pour la suite :

[154] Cet axiome a été présenté par SEN [1970], lequel est une version généralisée et faible d'un axiome (*Suppes' Grading Principle*) de Suppes (1966) portant sur des profils généralisés, formés d'une population réduite à deux individus.

Une relation binaire de préférence <u>étendue</u> $\succcurlyeq$ sur $S \times I_e$ est **inter-solution transitive**, si et seulement s'il existe un préordre partiel $\succcurlyeq'$ sur $S \times I_e$ tel que :

$$(x, i)\ \alpha_{\succcurlyeq}\ (y, j) \Leftrightarrow (x, i)\ \alpha_{\succcurlyeq'}\ (y, j),$$
$$\forall \text{ couple de positions } ((x, i), (y, j)) \in \{((z, k), (t, l)) \in (S \times I_e)^2 \text{ tel que : } z \neq t\}$$

Par exemple, la relation étendue (*a*) de la Figure 34 est inter-solution transitive. Par contre les relations (*b*) et (*c*) ne le sont pas.



Remarque : Les arcs réflexifs ont été omis.

Figure 34 : Illustration de l'inter-solution transitivité des relations binaires de préférence étendues.

Cette propriété effectue une bipartition de l'ensemble des couples de positions $(S \times I_e)^2$, en les couples de positions interindividuelles pour chaque solution, dont les jugements sont quelconques, et les couples de positions inter-solutions qui doivent être une partie d'un préordre étendu.

Une règle d'agrégation simple sur $S \times I_e$ **conserve l'inter-solution transitivité**, si l'image d'un profil de relations binaires de préférence inter-solution transitives sur $S \times I_e$, par cette règle d'agrégation est une relation binaire de préférence inter-solution transitive.

La règle d'équité de Suppes-Sen conserve la transitivité. De plus, elle reste transitive dans certains cas où la relation individuelle binaire de préférence étendue ne l'est pas. Nous avons le résultat suivant :

Proposition 11 : Étant donnée une relation binaire de préférence étendue $\succcurlyeq$ sur $S \times I_e$, avec $\mathrm{card}(S) > 2$. Si $\succcurlyeq$ est inter-solution transitive, alors son image $\succcurlyeq_{\mathrm{SSE}}$ par la règle d'équité de Suppes-Sen SSE est transitive.

Preuve : Nous devons montrer que $\succcurlyeq_{\mathrm{SSE}}$ est transitive, c'est-à-dire que tout triplet de solutions $x, y, z \in S$ avec $x \neq y \neq z$ vérifiant $x \succcurlyeq_{\mathrm{SSE}} y$ et $y \succcurlyeq_{\mathrm{SSE}} z$, vérifie aussi $x \succcurlyeq_{\mathrm{SSE}} z$. Si $\mathrm{card}(I_e) = \{i\}$ alors la relation est vérifiée. Dans le cas général, nous avons :

$x \succcurlyeq_{\mathrm{SSE}} y$ implique qu'$\exists$ une permutation $\sigma_{(x, y)}$ telle que : $(x, i) \succcurlyeq (y, \sigma_{(x, y)}(i))\ \ \forall\ i \in I_e$, et

$y \succcurlyeq_{\mathrm{SSE}} z$ implique qu'$\exists$ une permutation $\sigma_{(y, z)}$ telle que : $(y, i) \succcurlyeq (z, \sigma_{(y, z)}(i))\ \ \forall\ i \in I_e$.

Comme $\succcurlyeq$ est inter-solution transitive, alors il existe un préordre $\succcurlyeq'$ sur $S \times I_e$ vérifiant :

$$(w, i)\ \alpha_{\succcurlyeq}\ (t, j) \Leftrightarrow (w, i)\ \alpha_{\succcurlyeq'}\ (t, j),\ \forall\ ((z, i), (t, j)) \in (S \times I_e)^2 \text{ tel que : } z \neq t.$$

Sachant de plus que $x \neq y \neq z$, alors le préordre $\succcurlyeq'$ vérifie :

$$(x, i) \succcurlyeq' (y, \sigma_{(x, y)}(i))\ \ \forall\ i \in I_e, \quad \text{et} \quad (y, i) \succcurlyeq (z, \sigma_{(y, z)}(i))\ \ \forall\ i \in I_e.$$

En tant que préordre, $\exists$ alors une permutation $\sigma_{(x, z)}$ telle que : $(x, i) \succcurlyeq' (z, \sigma_{(x, z)}(i))\ \ \forall\ i \in I_e$. Par exemple, $\sigma_{(x, z)} = \sigma_{(y, z)} \circ \sigma_{(x, y)}$, avec $\circ$ l'opérateur de composition sur l'ensemble des permutations. La permutation $\sigma_{(y, z)} \circ \sigma_{(x, y)}$ vérifie $(x, i) \succcurlyeq' (z, \sigma_{(y, z)}(\sigma_{(x, y)}(i)))\ \ \forall\ i \in I_e$, du fait de la transitivité de $\succcurlyeq'$. Par conséquent, comme $x \neq z$, alors $(x, i) \succcurlyeq (z, \sigma_{(x, z)}(i))\ \ \forall\ i \in I_e$. Ce qui démontre donc que $x \succcurlyeq_{\mathrm{SSE}} z$. ❑

La règle SSE peut être appliquée à des relations binaires de préférence étendues $\succcurlyeq$ ne respectant pas l'inter-solution transitivité. Par exemple, la Figure 34 (*b*) montre une telle relation, pour laquelle il existe une permutation satisfaisant $(y, i) \succ (x, \sigma(i))\ \forall\ i \in I_e$. D'un autre coté,

cette règle n'existe pas pour toute relation binaire de préférence étendue. Pour l'exemple (*c*) de la Figure 34, nous avons : $y \succ_{\text{SSE}} x$ et $x \succ_{\text{SSE}} y$. Ce qui est incohérent.
Dans ce cas nous pouvons définir une règle généralisant la règle SSE à tout élément de $\mathcal{R}(\text{RBP}, S \times I_e)$, et appelée règle d'équité de Suppes-Sen étendue ESSE :

La **règle d'équité de Suppes-Sen étendue** ESSE définit une relation binaire de préférence $\succcurlyeq_{\text{ESSE}}$ sur S, à partir d'une relation binaire de préférence étendue $\succcurlyeq$, comme suit : Considérons deux solutions x et y de S.

$x \succcurlyeq_{\text{ESSE}} y \quad \Leftrightarrow \quad$ il existe une permutation σ des individus de I_e telle que :
pour tout $i \in I_e$ $(y, i) \succcurlyeq (x, \sigma(i))$;

La préférence stricte $\succ_{\text{ESSE}}$ correspond à la partie asymétrique de $\succcurlyeq_{\text{ESSE}}$ et l'indifférence $\simeq_{\text{ESSE}}$ à la partie symétrique.

Les deux règles d'agrégation SSE et ESSE se confondent lorsque SSE est défini. Pour l'exemple (*c*) de la Figure 34, nous obtenons : $y \simeq_{\text{ESSE}} x$.

6.3.2.2. Maximin et Leximin

Rawls [1971] proposa une autre règle d'agrégation véhiculant un principe de justice : la règle du Maximin. Sa définition est la suivante :

La **règle du Maximin** (ou Principe de Différence de Rawls) définit une relation binaire de préférence $\succcurlyeq_{\text{MAX}}$ sur S, à partir d'une relation binaire de préférence étendue $\succcurlyeq$, comme suit : Considérons $x, y \in S$, tel qu'il existe un plus petit élément sur $\{x\} \times I_e$ pour le préordre partiel étendu $\succcurlyeq$ noté[155] $\min_{i \in Ie}(x, i)$, de même pour y. Alors,

$$x \succcurlyeq_{\text{MAX}} y \quad \Leftrightarrow \quad \min_{i \in Ie}(x, i) \succcurlyeq \min_{i \in Ie}(y, i) ;$$

La partie asymétrique de $\succcurlyeq_{\text{MAX}}$ correspond à la préférence stricte $\succ_{\text{MAX}}$ et la partie symétrique à l'indifférence $\simeq_{\text{MAX}}$.

C'est la position de la personne la plus mal lotie qui permet de juger de la qualité d'une solution. Lorsque le préordre étendu $\succcurlyeq$ admet un plus petit élément pour toute solution x (comme c'est le cas quand $\succcurlyeq$ est complet), alors la relation $\succcurlyeq_{\text{MAX}}$ est structurée en préordre complet. Dans le cas partiel, $\succcurlyeq_{\text{MAX}}$ est un préordre partiel (conservation de la transitivité), un plus petit élément n'existant pas toujours. La définition de cette règle peut être adaptée à la relation binaire de préférence étendue, non nécessairement transitive, si le plus petit élément sur $\{x\} \times I_e$ est remplacé par la position de meilleure qualité (voir § 3.4) pour $(\{x\} \times I_e, \succcurlyeq^{\text{inv}})$, avec $\succcurlyeq^{\text{inv}}$ l'inverse de $\succcurlyeq$ (i.e. $(x, i) \succcurlyeq^{\text{inv}} (x, j) \Leftrightarrow (x, j) \succcurlyeq^{\text{inv}} (x, i)$ (voir § 3.3.1.1)).

En remplaçant le plus petit élément par le plus grand élément dans la définition de la règle du Maximin, nous obtenons une nouvelle règle appelée **règle du Minimax**.

La règle du Maximin n'est pas une extension de la règle de Pareto, mais respecte l'unanimité de la préférence stricte U[$\succ$] (SEN [1970]). La règle du Maximin n'est donc pas non plus une extension de la règle SSE (SEN [1970, corollaire 9*5.1]). Par contre, la règle d'équité de Suppes-Sen et la règle de Pareto sont incluses dans la règle du Maximin (SEN [1970, théorème 9*5]).

SEN [1970] propose une version lexicographique de la règle du maximin lorsque la relation $\succcurlyeq$ est un préordre complet. Avant de la présenter, considérons pour chaque solution x de S une application lex_x de $\{1, \ldots, \text{card}(I_e)\}$ dans I_e vérifiant : $k_1 < k_2$ si et seulement si $(x, lex_x(k_2)) \succcurlyeq (x, lex_x(k_1))$. Cette application range par ordre croissant les individus, du moins bien loti au mieux loti dans la solution x, en tranchant arbitrairement entre deux individus ex æquos.

[155] Ce qui est équivalent à dire que $\min_{i \in Ie}(x, i) \in B(\{x\} \times I_e, \preccurlyeq) \neq \varnothing$, en utilisant la notation du § 3.4.1.

La **Règle du Leximin** (ou Extension Lexicographique du Principe de Différence de Rawls) définit une relation binaire de préférence $\succcurlyeq_{\text{LEX}}$ sur S, à partir d'un préordre total étendu $\succcurlyeq$, comme suit : Soit $x, y \in S$. Alors,

$$x \succ_{\text{LEX}} y \iff \text{il existe un } k \in \{1, \ldots, \text{card}(I_e)\} \text{ tel que : } \begin{cases} (x, lex_x(l)) \simeq (y, lex_y(l)) & \text{pour tout } 1 \leq l < k \\ (x, lex_x(k)) \succ (y, lex_y(k)) \end{cases}$$

$$x \simeq_{\text{LEX}} y \iff \text{pour tout } k \in \{1, \ldots, \text{card}(I_e)\},\ (x, lex_x(k)) \simeq (y, lex_y(k)).$$

Les préférences collectives de la règle du Leximin sont un préordre total sur S (conservation de la transitivité). Cette règle est une extension des règles de Pareto, du Maximin, (SEN [1970, p. 138]) et de SSE (HAMMOND [1979]). La Figure 35 récapitule quelques relations entre les différentes règles d'agrégation présentées. La règle du Leximin peut se généraliser aux relations binaires de préférence complètes $\succcurlyeq$ où la restriction à $\{x\} \times I_e$ est un préordre complet, pour toute solution x de S.

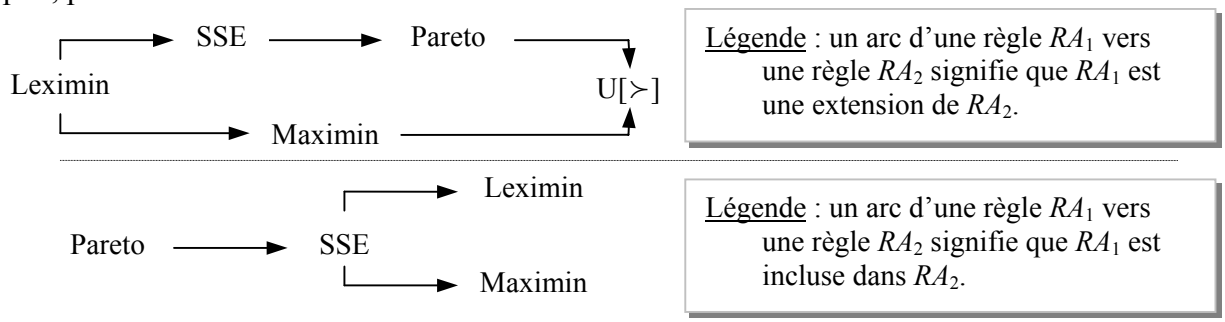


Figure 35 : Récapitulation des relations d'extension et d'inclusion entre les règles d'agrégation véhiculant des principes d'équité et d'efficacité.

Les règles du Maximin et du Leximin ont été très critiquées comme règles véhiculant des principes de justice et d'équité (SEN [1980], [1992], SUZUMURA [1997] parmi d'autres), car elles ne font aucun cas du nombre d'individus dont les intérêts sont lésés dans la défense des plus mal lotis. C'est la position la plus désavantagée qui fait la loi, et ces règles ne cherchent pas à savoir si cette position est mise en balance avec les intérêts d'une personne, d'un million ou d'un milliard de personnes. Par conséquent, des tentatives de définition de règles d'agrégation intermédiaires entre SSE et le Leximin ont été effectuées. Elles reposent sur des axiomes qui sont présentés ci-dessous.

6.3.2.3. Versions Ordinales du Principe de Transfert de Pigou-Dalton

Une axiomatisation de la règle du Leximin utilise l'axiome d'équité de HAMMOND, non vérifié par SSE, dont la définition est la suivante[156].

Axiome HE (Axiome d'Équité de Hammond) : Une règle d'agrégation étendue RA vérifie l'axiome HE si, pour tout couple de solutions x et y de S,

$$\left.\begin{array}{l} \text{il existe } i \text{ et } j \text{ de } I_e \text{ tels que : } (y,i) \succ (x,i) \succ (x,j) \succ (y,j) \text{ et} \\ (x,k) \simeq (y,k) \quad \text{pour tout } k \in I_e \setminus \{i,j\} \end{array}\right\} \Rightarrow x \succ_{RA} y.$$

Cet axiome est illustré à la Figure 36 (*a*). Remarquons une certaine similitude avec le principe de Pigou-Dalton (voir Figure 27).

[156] Ce résultat a été obtenu par HAMMOND [1979] sous l'hypothèse d'existence d'un vrai-critère étendu formant une extension de la relation binaire de préférence étendue d'entrée. Différents systèmes axiomatiques ont été proposés dans la littérature pour caractériser la règle du Leximin. SEN [1986, § 6.3] donne un récapitulatif formé de 7 caractérisations différentes.

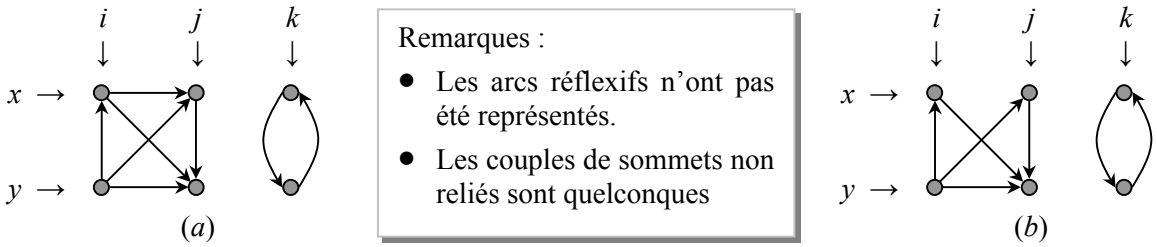


Figure 36 : Illustration des axiomes d'équité de Hammond (*a*) et de transfert de Foster-Sen (*b*).

L'axiome HE peut être relaxé, en n'imposant pas de préférence stricte entre (x, i) et (x, j), (inspiré de FOSTER & SEN [1997, p. 216]), comme le montre la Figure 36 (*b*). Nous obtenons une version ordinale du principe de transfert de Pigou-Dalton appelé **principe de transfert de Foster-Sen** :

Axiome FST (Principe de Transfert de Foster-Sen) : Une règle d'agrégation étendue *RA* vérifie l'axiome FST si, pour tout couple de solutions x et y de S,

$$\left.\begin{array}{l} \text{il existe } i \text{ et } j \text{ de } I_e \text{ tels que :} \left|\begin{array}{l}(y,i) \succ (x,k) \\ (x,k) \succ (y,j)\end{array}\right. \text{ pour } k \in \{i,j\}, \text{ et} \\ (x,k) \simeq (y,k) \quad \text{pour tout } k \in I_e \setminus \{i,j\} \end{array}\right\} \Rightarrow x \succ_{RA} y.$$

Par construction, nous avons les implications suivantes : HE $\Rightarrow$ FST. En d'autres termes, toute règle d'agrégation vérifiant l'axiome HE vérifie aussi l'axiome FST. Les règles d'agrégation, autre que la règle du Leximin, satisfaisant le principe de Suppes, l'indifférence de Suppes et l'axiome FST, forment des candidats attrayants pour évaluer l'équité couplée à l'unanimité (*fairness*). Pour le moment aucune règle alternative allant dans ce sens n'a été proposée. Pour plus d'investigations, nous renvoyons aux articles précités ainsi qu'à SEN [1977*a*], et plus récemment D'ASPREMONT & GEVERS [2002] et BLACKORBY *et al*. [2002]. En attendant, nous nous limitons à la seule règle d'équité de Suppes-Sen, acceptée unanimement comme règle véhiculant des principes minimaux d'équité et d'efficacité pour synthétiser les préférences ordinales étendues d'un individu arbitraire.

6.3.3. Efficacité et Équité à partir d'un Profil Quelconque de Relations Binaires de Préférence Étendues

Plaçons-nous maintenant dans le cadre d'un profil quelconque de relations binaires de préférence étendues $\mathcal{F}(I_{se}, I_e, S)$, où I_{se} est l'ensemble des émetteurs, I_e l'ensemble des individus observés, et S l'ensemble des solutions donné en extension ou non. En TCS, différentes manières d'agréger ce type de profil ont été utilisées (Figure 37).

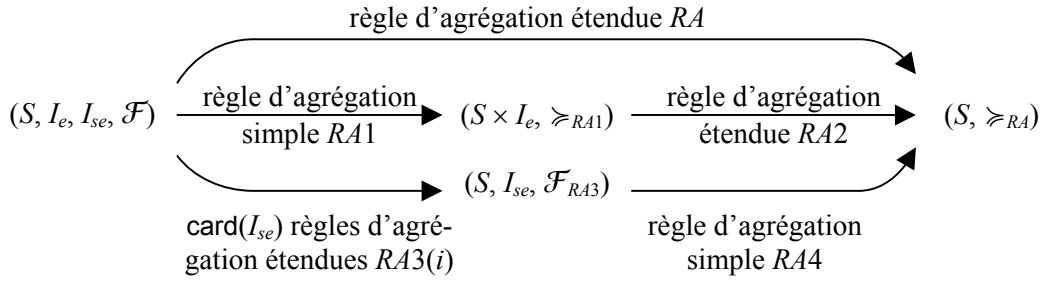


Figure 37 : Différents types d'agrégations exploités en TCS pour traiter les profils quelconques de relations binaires de préférence étendues.

Le premier type d'agrégation utilise une unique règle d'agrégation étendue *RA*. Dans le cas d'un profil généralisé ($\Leftrightarrow I_{se} = I_e$), rien n'empêche d'utiliser une règle d'agrégation simple, sachant que tout profil généralisé contient l'information d'un profil simple via la relation générale **(**4-14**)** page 129 qui, en termes de relation binaire de préférence donne :

$$x \succcurlyeq'_i y \quad \Leftrightarrow \quad (x, i) \succcurlyeq_i (y, i), \text{ pour tout } (x, y, i) \in S \times S \times I_e,$$

avec $\succcurlyeq'_i$ (resp. $\succcurlyeq_i$) la relation binaire de préférence simple (resp. étendue) de l'individu *i*.

Nous présentons au § 6.3.3.3 un type de règle d'agrégation directe véhiculant un principe d'équité prometteur (la notion d'absence de convoitise), dont l'apparition est très récente en TCS.

Deux autres types d'agrégations ont été étudiés. Il s'agit de hiérarchies agrégatives à deux niveaux. Un premier type consiste à : (*a*) utiliser une règle d'agrégation <u>simple</u> *RA*1 transformant le profil quelconque de préférences individuelles étendues, en une relation binaire de préférence étendue ($\Leftrightarrow RA1(S \times I_e, I_{se}, \mathcal{F}) \in \mathcal{R}(\text{RBP}, S \times I_e)$), pour être ensuite (*b*) agrégée par une règle d'agrégation <u>étendue</u> *RA*2. Par exemple, *RA*1 peut correspondre à la règle de Pareto sur l'ensemble des positions, et *RA*2 peut être la règle IROR (§ 4.2.1.3) ou l'une de celles présentées au § 6.3.2. Lorsque *RA*1 est la règle de Pareto, elle vérifie l'axiome d'identité complète (§ 4.1.3.3), mais fournit des préférences collectives généralement très pauvres en information préférentielle. Par conséquent, d'autres règles sont conçues. Nous présentons au § 6.3.3.2 des travaux originaux dans ce cadre, aboutissant à un conflit entre l'unanimité et une conception de la liberté individuelle.

Le second type d'agrégations (*a*) synthétise les préférences de chaque individu émetteur *i* en une relation binaire de préférence sur *S*, par l'intermédiaire d'une règle d'agrégation étendue pouvant être individualisée *RA*3(*i*). (*b*) Le profil de relations binaires de préférence sur *S* obtenu est ensuite synthétisé avec une règle simple *RA*4. Nous illustrons au § 6.3.3.1, ce type d'agrégation avec la présentation d'un conflit entre l'unanimité et l'équité de Suppes-Sen.

6.3.3.1. Conflit entre Unanimité et Équité de Suppes-Sen

Lorsque le profil de type sympathie étendue $\mathcal{F}$ à notre disposition regroupe les préférences de plus d'un individu, les conflits sont encore plus fréquents. Ainsi, la règle d'équité de Suppes-Sen $\succcurlyeq_{\text{SSE}(i)}$ appliquée aux préférences étendues d'un individu *i* quelconque du profil (au lieu de l'émetteur arbitraire, dans le cas précédent) entre en conflit avec la règle associée à l'unanimité de la préférence stricte $\succcurlyeq_{\text{U}[\succ]}$ (SEN [1970, théorème 9*2]) ! Prenons l'exemple de la Figure 38 (Tiré de SEN [1970]).

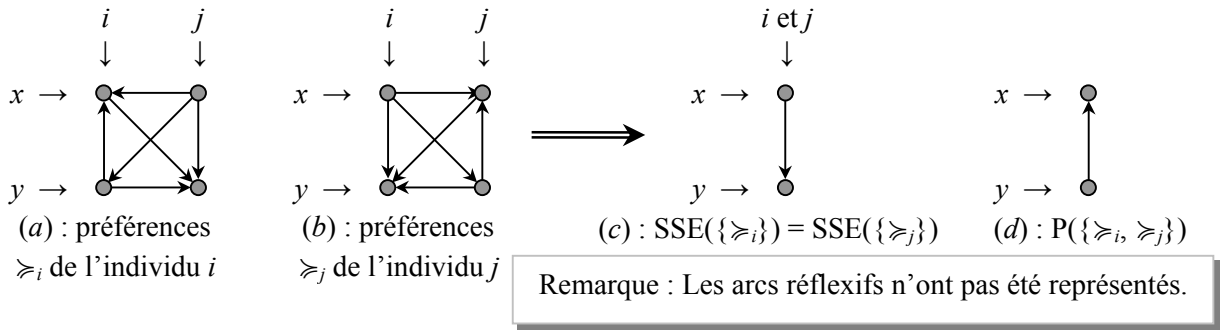


Figure 38 : Exemple de conflit entre l'unanimité de la préférence stricte et l'équité de Suppes-Sen.

Par conséquent, toute extension de l'unanimité $\succcurlyeq_{\text{U}[\succ]}$, comme la règle de Pareto, est incompatible avec $\succcurlyeq_{\text{SSE}(i)}$. Ce conflit est du même ordre que celui entre la règle de

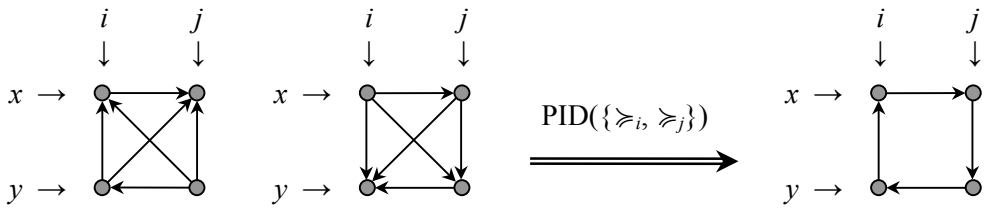
l'amendement et la majorité (§ 4.2.2.3), où les modèles préférentiels hiérarchiques peuvent ne plus conserver certaines propriétés fondamentales.

6.3.3.2. Conflits entre Liberté Individuelle et Unanimité

Une des formes que peut prendre la liberté individuelle est de considérer que chaque individu possède une *sphère personnelle reconnue*, dans laquelle sa préférence et elle seule compte pour déterminer la préférence sociale. Nous pouvons ainsi envisager que les choix effectués par un individu i sur les positions le concernant uniquement, c'est-à-dire sur les couples de positions $\{((x, j), (y, k)) \in (S \times \{i\})^2\}$, forment sa sphère personnelle reconnue. Dans ce cas, la liberté individuelle est satisfaite si l'axiome d'identité l'est. Par conséquent, la **règle de Pareto conditionnée par l'Identité** est une règle d'agrégation sur un profil de relations binaires de préférence sur $S \times I_e$, dont les préférences collectives sont une relation binaire de préférence <u>étendue</u> $\succcurlyeq_{\text{PID}}$ vérifiant :

$$(x, i) \succcurlyeq_{\text{PID}} (y, j) \Leftrightarrow \begin{cases} (x,i) \succcurlyeq_i (y,j) & \text{si } i = j \\ (x,i) \succcurlyeq_{\text{P}} (y,j) & \text{sinon.} \end{cases}$$

La règle d'agrégation de Pareto conditionnée par l'identité est une règle d'agrégation simple sur l'ensemble des positions. Elle se décompose en deux parties séquentielles : La première transforme le profil initial de préférences individuelles en profil respectant l'identité, en remplaçant l'avis de l'individu j par celui de i sur les couples de positions $\{((x, i), (y, i)) \in (S \times I_e)^2$, pour tout $x, y \in S\}$ et ce pour tout $(i, j) \in I_e \times I_{se}$. La seconde partie consiste à appliquer la règle de Pareto sur le profil modifié. Cette règle d'agrégation respecte donc en quelque sorte l'axiome d'identité ainsi que la règle de Pareto sur l'ensemble des positions. Par contre, elle ne conserve pas la transitivité. Prenons un exemple avec $S = \{x, y\}$ et $I_e = \{i, j\}$:



(*a*) : préférences $\succcurlyeq_j$ de l'individu *j*. (*b*) : préférences $\succcurlyeq_i$ de l'individu *i*. (*c*) : préférences collectives $\succcurlyeq_{\text{PID}}$

Remarque : Les arcs réflexifs ont été omis.

Figure 39 : Illustration de la non conservation de la transitivité par la règle d'agrégation de Pareto conditionnée par l'Identité.

Le principe de liberté individuelle exprimée par l'axiome d'identité paraît plutôt légitime. Et pourtant PID peut aboutir à des circuits de préférences strictes. On peut aller plus loin et constater que la règle de Pareto conditionnée par l'identité ne conserve pas non plus l'intersolution transitivité, et ce même lorsque S et I_e sont limités à deux éléments. La Figure 40 montre un exemple d'intransitivité de cette règle. Cet exemple illustre bien que même l'axiome d'unanimité de la préférence stricte U[$\succ$] couplé au principe de liberté individuelle exprimé par l'axiome d'identité, conduit à des préférences collectives incohérentes. Ce résultat est une version du fameux théorème d'impossibilité du Paretien libéral (ou paradoxe de Sen)[157]. Ce dilemme dit qu'il existe des situations où un principe doit être sacrifié pour l'autre.

[157] En TCS, l'unanimité de la préférence stricte est aussi appelée principe de Pareto faible. Pour un approfondissement de ce résultat voir SEN [1970], [1982*a*], [1983].

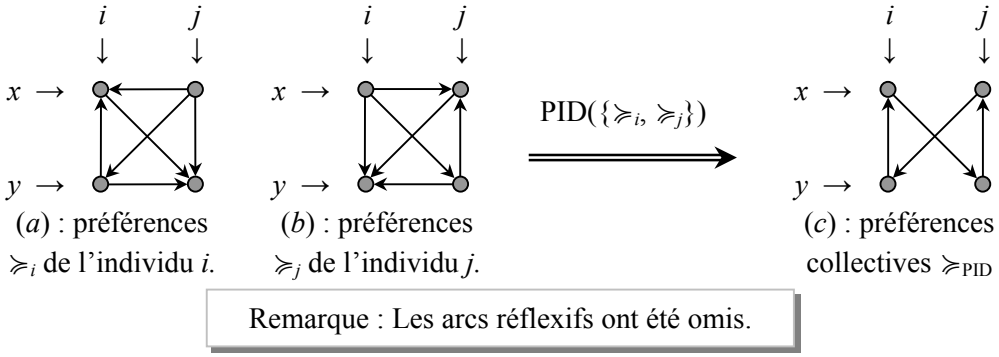


Figure 40 : Illustration de la non conservation de l'inter-solution transitivité par la règle d'agrégation de Pareto conditionnée par l'Identité.

Ce paradoxe informe que des circuits de préférences strictes peuvent être obtenus pour des règles satisfaisant à la fois un principe de liberté et un principe d'efficacité. Bien sûr ces intransitivités peuvent être traitées en utilisant des axiomes de choix adéquats (voir § 3.4), de manière à présenter aux décideurs les points conflictuels. Mais dans un cadre implicite, nous ne possédons pas encore d'outils de traitement adéquats. Par conséquent, les règles reposant sur de tels principes mais affaiblis, et qui, de plus, fournissent des préférences collectives transitives sont recherchées.

Beaucoup de travaux ont porté sur la définition d'une règle usant d'une forme affaiblie de la règle de Pareto, de la liberté ou/et d'une restriction du profil initial. Voir SEN [1983] pour une revue, ainsi qu'une réflexion très poussée sur la manière d'envisager la liberté pour aboutir à des solutions acceptables.

Remarquons que tout profil transitif n'aboutit pas toujours à des préférences collectives, si l'on applique PID suivi de SSE. Ainsi, les préférences collectives de l'exemple de la Figure 40 est une relation binaire de préférence étendue obtenue en appliquant PID. Lorsque ces préférences sont utilisées pour être agrégées par SSE, on aboutit à une incohérence. Il est alors nécessaire d'utiliser la version étendue ESSE (§ 6.3.2.1).

6.3.3.3. Équité Vue Comme une Absence de Convoitise

L'équité peut être vue comme une absence de convoitise des individus de I_e (voir SUZUMURA [1983], [1997] par exemple).

Par définition, l'individu i **envie** un autre individu j en une solution x selon le profil généralisé $\mathcal{F}$, noté $i \succ_{\mathrm{ENVY}(x)} j$, si et seulement si $(x, j) \succ_i (x, i)$, où $i, j \in I_e$ et $x \in S$. Cette relation binaire $\succ_{\mathrm{ENVY}(x)}$ sur I_e, permet alors de définir une relation binaire de préférence $\succcurlyeq_{\mathrm{ENVY}}$ sur S par :

Pour tout couple de solutions $x, y \in S$, $\quad x \succcurlyeq_{\mathrm{ENVY}} y \Leftrightarrow$

$$\{(i,j) \in I_e \times I_e \text{ tel que : } i \succ_{\mathrm{ENVY}(x)} j\} \subseteq \{(i,j) \in I_e \times I_e \text{ tel que : } i \succ_{\mathrm{ENVY}(y)} j\}$$

Intuitivement, la relation $\succcurlyeq_{\mathrm{ENVY}}$ informe que si i envie j en x selon $\mathcal{F}$, alors i reste envieux de j en y selon $\mathcal{F}$. L'occurrence de l'envie, parmi les individus, en y sous $\mathcal{F}$ n'est pas moins fréquent qu'en x sous $\mathcal{F}$. Dans ce sens, $x \succcurlyeq_{\mathrm{ENVY}} y$ signifie que x est au moins aussi équitable que y.

La règle d'agrégation $\succcurlyeq_{\mathrm{ENVY}}$ est un préordre sur S, et la règle de choix collectif associée identifie l'ensemble $M(S, \succcurlyeq_{\mathrm{ENVY}})$ des solutions pour lesquelles il n'existe aucune solution plus équitable, appelé aussi ensemble maximal (§ 3.4.1).

Cette manière de concevoir l'équité ouvre un très grand champ de recherche qui a été très peu étudié en théorie du choix social. Mettons quand même un bémol, car cette forme

d'équité peut entrer en conflit avec la règle de Pareto[158] : $M(S, \succcurlyeq_{\text{ENVY}}) \cap M(S, \succcurlyeq_{\text{P}}) = \varnothing$ pour certains couples $(S, \mathcal{F})$.

Point positif : Lorsque nous comparons cette règle d'agrégation étendue à la règle d'équité de Suppes-Sen (§ 6.3.3.2), nous remarquons que l'envie fait usage uniquement de l'information préférentielle intra-solution, tandis que SSE fait intervenir uniquement l'information préférentielle inter-solution. En d'autres termes, ces deux types de règles représentent deux aspects de l'équité qui peuvent être exploitées conjointement.

Les travaux sur l'agrégation efficace et équitable de profils de relations binaires de préférence étendues sont à leur début. Un éventail de recherches restent encore à mener.

6.3.4. Théorie des Jeux et Problèmes de Répartition

La théorie des jeux a aussi largement contribué à l'élargissement du problème de répartition pure, en concevant notamment des procédures interactives de choix de la solution réalisable la plus équitable. Le dialogue entre les acteurs est mis en avant[159]. Certaines procédures sont d'ailleurs basées sur l'unique dialogue entre les intéressés (la population I_e) sans intervention aucune de l'extérieur.

Plusieurs problèmes étudiés n'émettent aucune hypothèse sur les individus de la population (homogénéité de la population par exemple). Par conséquent, l'axiome d'égalité des proportions est remis en question pour une ouverture sur d'autres situations où l'équité est recherchée. De plus, les ressources peuvent ne pas être homogènes (par exemple un gâteau non homogène). Le but est d'arriver au choix d'une solution de compromis. Ces procédures interactives n'ont pas toujours de base axiomatique. Ces procédures interactives s'apparentent exactement aux procédures d'auto-planification utilisées pour résoudre certains problèmes d'élaboration de plannings infirmiers (§ 2.2.4.4).

La théorie des jeux coopératifs, elle, s'est intéressée à l'aspect axiomatique. Voir la synthèse de MOULIN & THOMSON [1997]. Mais il reste encore beaucoup à faire sur le sujet.

Les approches de la théorie des jeux et de la théorie du choix social sont complémentaires, et permettent de concevoir des outils d'aide à la décision plutôt satisfaisants pour ce type de problèmes. De plus elles simulent l'imagination pour la conception de systèmes interactifs d'aide à la décision dédiés à des problèmes plus complexes comme notre problème d'élaboration de plannings.

6.3.5. Allocation de Ressources, Préférences Individuelles et Équité

Fonder une mesure de l'équité uniquement sur les préférences individuelles étendues, peut aboutir à un certain nombre de difficultés. Tout d'abord, cette branche de l'éthique souffre de sa jeunesse : des conflits (§ 6.3.3) ont été identifiés, retardant la conception de règles d'agrégation satisfaisantes du point de vue de l'équité. Ensuite, dans la pratique, le manque de sympathie entre individus de I_{se} peut conduire à des préférences collectives extrêmement pauvres en information préférentielle, si une règle unanime comme la règle de Pareto est utilisée suivie de la règle SSE (voir SEN [1992], [1993*a*]).

L'alternative que représente l'inégalité basée sur les ressources a conduit à une littérature abondante portant aussi bien sur les règles d'agrégation équitables (§ 6.2.3) que sur l'identification des attributs de ressources pertinents.

---

[158] Pour un exemple, voir SUZUMURA [1997].

[159] Le livre de ROBERTSON & WEBB [1998] est une synthèse assez complète sur le sujet. Il présente un certain nombre d'algorithmes de théorie de graphes, et de résultats de complexité. Voir aussi le compte rendu de ce livre effectué par SU [2000].

6.3.5.1. Espace d'Évaluation

L'inégalité multidimensionnelle, exprimée par les règles d'agrégation satisfaisant les axiomes UL, CIT et CIL, peut être utilisée pour obtenir une bonne évaluation de l'équité même lorsque la population est hétérogène. résoudre le problème que pose la diversité humaine (la population n'est plus homogène). Pour cela, il est nécessaire que les attributs des ressources soient exhaustifs. En effet, l'équité dans un espace d'évaluation décrit par un ensemble d'attributs (ou espace des critères) n'implique généralement pas l'équité dans un autre espace d'évaluation. Cet espace peut faire toute la différence. Par exemple, mesurer l'inégalité humaine en utilisant le salaire comme unique attribut peut aboutir à des jugements complètement biaisés. L'être humain est caractérisé par sa diversité. Du fait que nous ayons des besoins, des aspirations, des aptitudes différents, un environnement personnel dissemblable, etc. font qu'à égalité dans un espace, nous soyons inégaux dans un autre[160].

L'identification de tous les facteurs (ou attributs) susceptibles d'influencer cette évaluation peut se faire en consultant les concernés. Cette énumération doit être réalisée périodiquement, car les avis évoluent au cours du temps[161].

En plus de l'ensemble exhaustif des attributs pertinents, les acteurs peuvent intervenir pour affiner la comparaison partielle existant entre ces attributs. Le paragraphe suivant y est consacré.

6.3.5.2. Comparabilité Partielle d'Attributs de l'Espace d'Évaluation

Au lieu d'utiliser directement la règle de Pareto sur le profil $\mathcal{F}(S \times I_e, R)$:

$$(x, i) \succcurlyeq (y, j) \Leftrightarrow \forall\, r \in R,\ x(r, i) \geq y(r, j)$$

ou encore la dominance de Lorenz généralisée multidimensionnelle (§ 6.3.1), SEN [1992] utilisa un autre stratagème : Un ensemble de personnes I_{se} est considéré. Chaque individu formule un avis sur les taux de substitution (ROY [1985], VINCKE [1989], ROY & BOUYSSOU [1993]) qu'il utiliserait pour agréger les ressources via une somme pondérée. Ainsi l'individu $k \in I_{se}$ considère que :

en une position (x, i) de $S \times I_e$, 1 unité de l'attribut de référence r_{ref} est égale à $w_k(r_{\text{ref}}, (x, i), r)$ unités de r, et ce pour tout $r \in R \setminus \{r_{\text{ref}}\}$.

Ces taux de substitution permettent par sommation, de définir un vrai-critère sur l'ensemble des positions par individu de I_{se}, définissant ainsi un autre profil $\mathcal{F}(S \times I_e, I_{se})$ de vrai-critères sur l'ensemble des positions. Ce profil est ensuite agrégé en utilisant la règle de Pareto pour obtenir un préordre partiel sur l'ensemble des positions $S \times I_e$. En d'autres termes :

$$(x, i) \succcurlyeq (y, j) \ \Leftrightarrow\ \forall\, k \in I_{se},\ \sum_{r \in R} w_k(r_{\text{ref}}, (x,i), r) \cdot x(r,i) \geq \sum_{r \in R} w_k(r_{\text{ref}}, (y,j), r) \cdot y(r,j)$$

c'est-à-dire, la règle de Pareto sur le profil de vrai-critères.

Comme fait remarquer SEN [1992], il est souvent possible d'établir une relation d'importance relative entre les variables de l'espace d'évaluation des positions (attributs des ressources). Par contre les avis divergent sur l'importance exacte en termes de taux de substitution, et ils évoluent avec le temps.

---

[160] Pour un approfondissement sur les différentes théories éthiques concernant les attributs pertinents pour mesurer le bien-être, la pauvreté, etc., nous renvoyons à SEN [1992]. L'approche théorique qui pour le moment a surmonté toutes les critiques en décision publique est l'approche « capabilité » (SEN [1980], [1987*a*], [1987*b*], [1992], [1993*a*], FOSTER & SEN [1997]). Elle a conduit à établir des mesures de bonne qualité du bien-être ; à mieux comprendre certaines famines historiques, la pauvreté dans les pays en voie de développement et dans les pays industrialisés, l'inégalité entre l'homme et la femme dans certains pays comme l'Inde. L'ONU s'en est inspiré pour confectionner les indicateurs internationaux utilisés dans son HDR (*Human Development Report* : http://hdr.undp.org/), qui est la base des programmes internationaux pour le développement humain, etc..

[161] Plus précisément, l'ensemble des attributs d'évaluation sur l'ensemble des positions devrait être une famille cohérente de critères, dans le même sens employé par ROY & BOUYSSOU [1993].

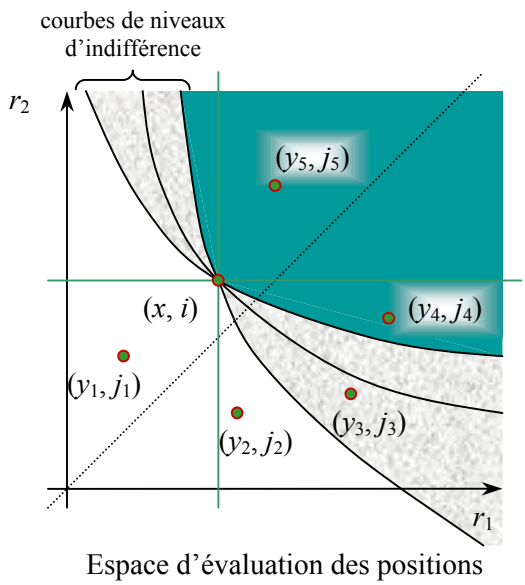


Figure 41 : Illustration de l'utilisation de préférences individuelles pour une meilleure évaluation des positions (Extrait de SEN [1992]).

La formule précédente s'illustre dans un cadre bidimensionnel et sous certaines hypothèses de continuité par la Figure 41. Si la règle de Pareto est directement appliquée sur $\mathcal{F}(S \times I_e, R)$, alors la partition par les attitudes fondamentales de l'espace d'évaluation des positions est donnée par les lignes horizontale et verticale passant par (x, i) . Nous pouvons donc dire que : (y_5, j_5) est préférée à (x, i), qui est préférée à (y_1, j_1) ; toutes les autres positions étant incomparables à (x, i). L'utilisation de l'avis des individus de I_{se}, (traduit par des courbes de niveaux des positions indifférentes à (x, i)) sur l'importance accordée aux attributs, permet de compléter la relation binaire de préférence étendue. Cet ajout permet alors de classer (y_2, j_2) et (y_4, j_4).

L'utilisation des taux de substitution, de la somme pondérée et de la dominance de Pareto aboutit à une relation binaire de préférence étendue plus précise que celle que l'on aurait obtenue en utilisant directement la règle de Pareto. Cette relation peut ensuite être agrégée par la règle SSE. Ainsi, il est possible de coupler à la fois l'information évaluationnelle basée sur les ressources multidimensionnelles et les préférences individuelles.

Cette méthode est généralisable à d'autres règles d'agrégation. La Figure 42 présente différentes hiérarchies agrégatives utilisables pour synthétiser l'information préférentielle des ressources multidimensionnelles. Elle n'est pas exhaustive.

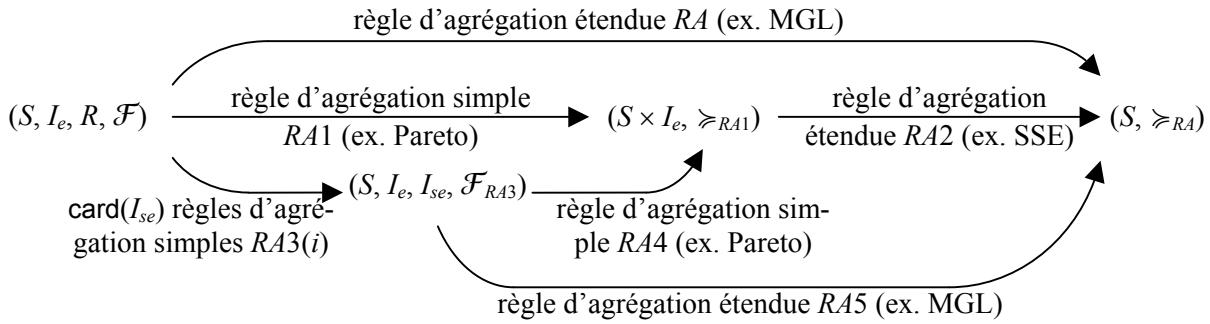


Figure 42 : Quelques hiérarchies agrégatives des préférences individuelles fournies par des quantités de ressources multidimensionnelles.

Les taux de substitution définissent une forme de corrélation partielle des attributs, qui n'est pas du tout pris en compte par la dominance de Lorenz généralisée multidimensionnelle (RA = MGL) ; cette étape s'effectuant de manière indépendante. Dans le cas particulier où les règles d'agrégation individuelles $RA3(i)$ définissent des préférences collectives structurées en vrai-critères, la dominance MGL peut être utilisée en tenant compte de cette information supplémentaire ($RA5$ = MGL). C'est le cas lorsque les taux de substitution et la somme pondérée sont utilisés.

D'autres combinaisons entre l'approche 'ressources' et l'approche 'préférences individuelles' sont possibles. Les acteurs peuvent intervenir à différents stades de la synthèse d'informations préférentielles. Les acteurs sont ainsi mieux intégrés dans le processus

d'évaluation, et par la suite dans tout le processus de décision. Ces interventions améliorent par la même occasion l'évaluation, rendant son verdict plus précis et l'utilisation des ressources dans l'évaluation plus acceptable.

6.4. Équité et Efficacité pour l'Évaluation des Plannings de Travail

Les différentes règles d'agrégation véhiculant des principes d'équité et d'efficacité, peuvent être utilisées à des fins agrégatives pour notre problème. Elles suggèrent d'autres manières de gérer certains conflits aussi bien personnels, qu'entre acteurs et groupes d'acteurs. Plusieurs hiérarchies agrégatives sont possibles. Nous introduisons sur des exemples cette nouvelle manière de modéliser et agréger les préférences sur les PEPTP.

6.4.1. Modèles Préférentiels des Problèmes d'Élaboration de Plannings de Personnel

Nos problèmes offrent l'opportunité d'étudier des profils comportant les préférences d'acteurs ayant différents rôles au sein du système (§ 1.1.2). Notons AC l'ensemble des acteurs. Ainsi, il existe trois types d'acteurs principaux, les employés E, les clients CL et les managers MG. Notons ACP l'ensemble de ces acteurs : $ACP = E \cup CL \cup MG \subseteq AC$. L'avis d'acteurs secondaires ACS sont aussi pris en considération : $AC = ACP \cup ACS$. Notons I_{PEPTP} une instance de PEPTP. Dans un cadre général, le modèle préférentiel des PEPTP comporte un grand nombre d'informations qui se résume comme suit :

- les préférences individuelles des employés E sur l'ensemble des plannings $S(I_{\text{PEPTP}}) \times E$;
- les préférences des clients CL, agrégés par des règles reflétant la stratégie clientèle de l'unité sur les plannings $S(I_{\text{PEPTP}}) \times CL$. Souvent, ce sont les responsables marketing MK du système qui émettent ces préférences, avec $MK \subseteq MG$;
- les préférences organisationnelles et les coûts formulés par les managers MG sur l'ensemble des plannings $S(I_{\text{PEPTP}}) \times ACP$;
- les préférences des émetteurs secondaires ACS sur les conditions des employés, des clients et des propriétaires $S(I_{\text{PEPTP}}) \times ACP$.

En résumé, le profil le plus général des PEPTP est de type sympathie étendue ($S(I_{\text{PEPTP}})$, ACP, AC, $\mathcal{F}$), où tous les acteurs AC s'expriment sur l'ensemble des conditions de chaque acteur principal $S(I_{\text{PEPTP}}) \times ACP$.

La liste des plannings étant généralement inconnue à cause de l'aspect combinatoire des PEPTP, l'information n'est pas fournie directement sur l'ensemble des positions $S(I_{\text{PEPTP}}) \times ACP$. Une grosse partie des préférences s'exprime dans un cadre où toute information extérieure est supposée égale par ailleurs (*ceteris paribus*). Par exemple, les préférences s'expriment entre les horaires de travail individuels, entre les combinaisons de propriétés que peuvent satisfaire un horaire ou un planning.

6.4.2. Agrégation Basée sur le Rôle des Acteurs Principaux

La masse d'information préférentielle à agréger est très importante. Comment faire usage des règles d'agrégation présentées dans les deux premières sections du chapitre pour obtenir une évaluation globale et fiable des plannings basée sur l'équité et l'efficacité ? Les possibilités sont multiples.

Les agrégations usuelles des problèmes d'élaboration de plannings considèrent trois types de critères d'agrégation (voir Figure 1) : la qualité de vie des travailleurs[162] induite par les

[162] En sciences humaines et sociales, la qualité de vie est recherchée pour les travailleurs (QWL = *quality of work life*). Ce critère regroupe divers indicateurs de sécurité, hygiène, etc., en plus des informations préféren-

plannings, la qualité de service aux clients induits par les plannings, et les préférences managériales. Nous les formalisons ci-dessous :

– La **qualité de vie des travailleurs induite par les plannings** QV correspond à l'agrégation partielle équitable et efficace des préférences des travailleurs E et des émetteurs secondaires ACS :

$$pref_{\mathrm{QV}}(E \cup ACS, S(I_{\mathrm{PEPTP}})) = \mathrm{QV}(\mathcal{F}(E \cup ACS, E, S(I_{\mathrm{PEPTP}}))) ;$$

– La **qualité de service induite par les plannings** QS est l'agrégation partielle des préférences étendues des clients, des préférences managériales et d'acteurs secondaires (associations de consommateurs, etc.) :

$$pref_{\mathrm{QS}}(CL \cup MK \cup ACS, S(I_{\mathrm{PEPTP}})) = \mathrm{QS}(\mathcal{F}(CL \cup MK \cup ACS, CL, S(I_{\mathrm{PEPTP}}))) ;$$

– Les **préférences managériales** CO sont le fruit de l'agrégation partielle des coûts et des préférences organisationnelles :

$$pref_{\mathrm{CO}}(MG, S(I_{\mathrm{PEPTP}})) = \mathrm{CO}(\mathcal{F}(MG, ACP, S(I_{\mathrm{PEPTP}})) ;$$

Ces trois critères généraux partiels peuvent être ensuite agrégés avec une règle d'agrégation finale RA, pour obtenir une évaluation globale des plannings :

$$\begin{aligned} pref(AC, S(I_{\mathrm{PEPTP}})) = RA(\{&pref_{\mathrm{QV}}(E \cup ACS, S(I_{\mathrm{PEPTP}})),\\ &pref_{\mathrm{QS}}(CL \cup MK \cup ACS, S(I_{\mathrm{PEPTP}})),\\ &pref_{\mathrm{CO}}(MG, S(I_{\mathrm{PEPTP}}))\}) \end{aligned}$$

La hiérarchie agrégative obtenue est notée RA(QV, QS, CO). La figure ci-dessous illustre cette hiérarchie via la relation d'agrégation (voir p. 140).

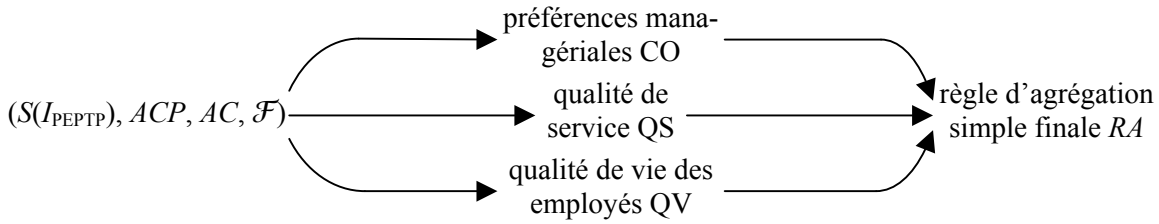


Figure 43 : Relation d'agrégation de la hiérarchie agrégative usuelle des PEPTP.

Bien sûr, ce choix d'agrégation est arbitraire. Mais elle reflète bien la pratique générale d'agrégation. A l'inverse des modèles usuels vus aux § 2.2.3 à 2.2.8, notre modèle repose sur des bases axiomatiques, et elle se veut minimale au sens où aucun principe non choisi ne vient « polluer » la définition de l'évaluation globale.

Les préférences managériales et la qualité de service induite par les plannings ont plusieurs préférences individuelles en commun lorsque les préférences des clients sont estimées par les responsables marketing. Dans la pratique, la charge CHG est directement utilisée, sa définition étant d'être un compromis entre la qualité de service et les préférences managériales. Elle s'exprime formellement par :

$$\begin{aligned} pref_{\mathrm{CHG}}(CL \cup MG \cup ACS, S(I_{\mathrm{PEPTP}})) = \mathrm{CHG}(\{&pref_{\mathrm{QS}}(CL \cup MK \cup ACS, S(I_{\mathrm{PEPTP}})),\\ &pref_{\mathrm{CO}}(MG, S(I_{\mathrm{PEPTP}}))\})\\ &\mathrm{CHG}_1(\mathcal{F}(CL \cup MG \cup ACS, CL \cup MG, S(I_{\mathrm{PEPTP}}))) ; \end{aligned}$$

La charge peut alors remplacer QS et CO dans la hiérarchie agrégative. Remarquons que les préférences collectives obtenues avec CHG ne sont pas toujours égales aux préférences collectives obtenues avec CHG_1. La règle CHG ne tient aucun compte de l'information préférentielle individuelle exprimée par les clients et acteurs secondaires sur les gestionnaires.

---

tielles propres aux plannings (SIRGY *et al.* [2001]). Par conséquent, nous précisons que QWL est induite par les plannings.

Remarquons que bien d'autres hiérarchies agrégatives sont possibles, comme par exemple celles présentées à la Figure 42. Selon la hiérarchie utilisée, certaines préférences individuelles sont entièrement utilisées, et d'autres sont partiellement écartées.

Nous n'envisageons pas ici de comparer les ensembles de choix obtenus avec différentes hiérarchies agrégatives. Toutes ces comparaisons ont un vive intérêt pour mieux comprendre et évaluer les plannings.

Dans la suite de cette section, nous entrons un peu plus dans les détails du modèle préférentiel des PEPTP, en étudiant des cas particuliers et des exemples. Tout d'abord, nous étudions la prise en compte d'une répartition équitable de la masse salariale entre les employés. Puis, nous approfondissons la question suivante : Comment utiliser la règle de Pareto et les axiomes d'équité présentés précédemment sur les préférences des travailleurs et des acteurs secondaires pour obtenir une mesure minimale de la qualité de vie ?

6.4.2.1. Répartition Équitable de la Masse Salariale

Parmi les problèmes identifiés au § 2.2, certains font usage d'un unique critère de coût, comme PEPTA(FL) ou les PEPTREG usuels associant un coût à chaque horaire. Si le coût d'un planning correspond uniquement à sa masse salariale, une modélisation plus juste chercherait aussi à répartir équitablement cette masse salariale entre les employés. Les nouveaux modèles préférentiels obtenus sont alors bicritère, le premier correspondant à la minimisation du coût total et le second étant la dominance de Lorenz.

Ces deux critères usent du vrai-critère étendu $f(x, h) = c(h)\cdot x(h)$ pour les PEPTREG usuels. Le coût total correspond à une somme simple de ces coefficients et le second effectue des comparaisons de Lorenz. Ces deux critères peuvent ensuite être agrégés de différentes manières. Par exemple, utiliser la règle lexicographique accordant la priorité au coût total, ou utiliser la dominance de Lorenz généralisée, etc.

Pour PEPTA(FL), la répartition équitable de la masse salariale peut engendrer des inégalités dans la répartition des horaires, car elle correspondrait à l'utilisation de la dominance de Lorenz sur le vrai-critère étendu $f(x, e) = \sum_{h\in H(e)} ct(e,h)\cdot x(e,h)$. En effet, si les employés ont un salaire de l'heure individualisé, la recherche d'une répartition équitable de la masse salariale entre les employés implique une inégalité de la durée de travail individuelle. Ainsi, pour arriver à une égalité de salaire, il est nécessaire que les moins payés travaillent plus en moyenne, même si cela n'est pas spécifié par les contrats de travail.

6.4.2.2. Équité et Efficacité dans la Conception des Préférences Individuelles

À un niveau individuel, la recherche d'une répartition équitable et efficace peut concerner plusieurs attributs pertinents des plannings. Par exemple :

- les horaires répartissant sur tout l'horizon de planification les matins, les soirs et les nuits sont mieux appréciés par un individu (JASZKIEWICZ [1997], [2001], JAUMARD *et al.* [1998], etc.) ;
- les horaires présentant des durées journalières (hebdomadaires, etc.) qui ne fluctuent pas trop sur l'horizon de planification sont mieux appréciés par un individu ;
- les horaires présentant des week-ends de repos bien distribués sur l'horizon de planification sont mieux appréciés par un individu ; etc..

Traitons l'exemple des préférences d'un travailleur e pour une répartition équitable des jours de repos tombant le week-end. Considérons donc :

- un horizon de planification comportant $n \geq 0$ week-ends successifs et travaillés (à quantité et qualité de travail individuel équivalentes), et

– un ensemble $\vartheta_{\text{week-end}}$ de situations souhaitées, associant à chaque week-end une situation « disposer de tout le week-end » ($\text{card}(\vartheta_{\text{week-end}}) = n$).

Exemple 6-1. Notre homme e énonce que :

« Pour un nombre m donné de week-ends ($0 \leq m \leq n$), les horaires préférés sont ceux, où les m week-ends sont bien distribués (répartis) tout le long de l'horizon, sans distinction (préférence) aucune entre les week-ends. »

Par exemple, prenons $(n, m) = (10, 5)$, et considérons les horaires h_1, h_2 et h_3 de e distribuant les 5 week-ends de libres selon le graphique suivant (Figure 44 (a)) :

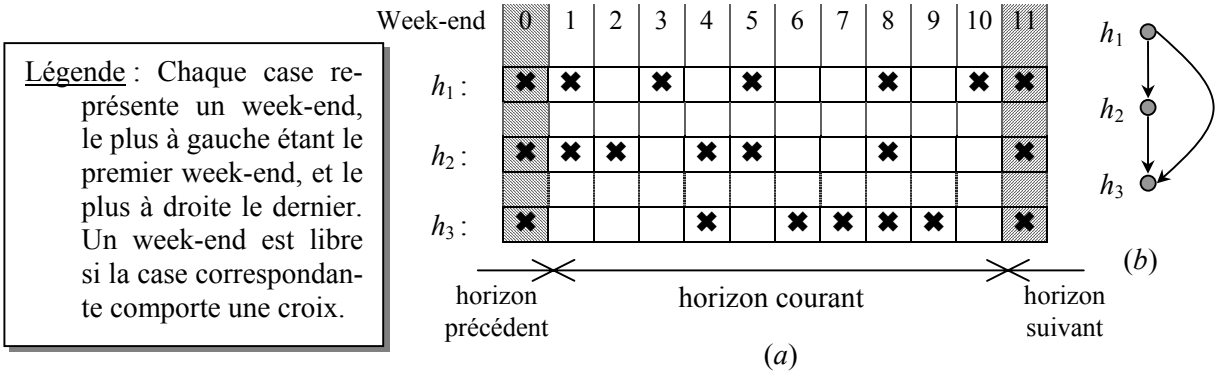


Figure 44 : Représentations de 3 horaires de travail qui diffèrent par leur répartition des week-ends de repos.

L'énoncé précédent peut s'interpréter de la façon suivante : l'individu e préfère l'horaire h_1 aux deux autres, et l'horaire h_2 à h_3. La Figure 44 (b) illustre cette relation par un graphe. Ce sont les préférences induites par les situations $\vartheta_{\text{week-end}}$ sur l'ensemble des horaires $H(e)$ de e.

Si cette relation est la seule qui influence les préférences de l'individu e, alors cette information peut être modélisée avec la dominance de Lorenz :

Notons $m(h)$ le nombre de week-ends de repos que compte l'horaire h. Désignons par $r(h, i)$ le rang qu'occupe le i^e week-end de repos de l'horaire h. Sur la Figure 44(a) le premier week-end de repos de l'horaire h_1 correspond au premier week-end de l'horizon de planification ($\Leftrightarrow r(h_1, 1) = 1$), tandis que pour l'horaire h_3 elle correspond au 4e week-end de l'horizon ($\Leftrightarrow r(h_2, 1) = 4$). La dominance de Lorenz s'applique sur le vrai-critère étendu u_{wer} défini sur l'ensemble des couples (h, i) formés d'un horaire h et d'un indice de week-end de repos i, comme suit :

$$u_{wer}(h, i) = r(h, i) - r(h, i-1), \qquad \text{avec } (h, i) \in H(e) \times \{1, \ldots, m(h)+1\}.$$

Pour le moment, nous supposons que $r(h, 0)$ désigne le dernier week-end l'horizon précédent ($\Leftrightarrow r(h, 0) = 0$) et que $r(h, m(h)+1)$ désigne le premier week-end de l'horizon suivant ($\Leftrightarrow r(h, m(h)+1) = n+1$). Alors les préférences de l'employé e prennent la forme du préordre partiel $\succcurlyeq_{L(wer),\, e}$ défini par :

$$h \succcurlyeq_{L(wer),\, e} h' \;\Leftrightarrow\; [L_{u_{wer}}(i / m(h)) \geq L_{u_{wer}}(i / m(h')),\ \text{quelque soit } i \in \{1, \ldots, m+1\}]\ ;$$

pour $h, h' \in H(e)$ et $m(h) = m(h') = m$. L'exemple précédent donne ainsi les courbes de Lorenz suivantes :

$11 \cdot L_{u_{wer}}(h_1) = (1, 2, 4, 6, 8, 11)$; $11 \cdot L(h_2) = (1, 2, 3, 5, 8, 11)$ et $11 \cdot L(h_3) = (1, 2, 3, 5, 7, 11)$.

et le préordre $\succcurlyeq_{L(wer)}$ correspond au graphe de la Figure 44(b), les arcs réflexifs n'ayant pas été représentés.

Lorsque le nombre de week-ends de repos varie d'un horaire à l'autre, l'axiome plutôt naturel utilisable est l'invariance sous les réplications de la population PRI (§ 6.2.6.1). Par con-

tre, notre cadre étant différent et spécifique, il est toujours pertinent de confronter cet axiome à l'intuition.

Bien entendu, l'énoncé préférentiel qui vient d'être présenté peut être enrichi, voire modifié par une autre information préférentielle émise par *e*, comme l'exemple suivant :

Exemple 6-2 :

« Quel que soit le nombre de week-ends où je ne suis pas d'astreinte, à choisir entre deux horaires où aucune ne domine l'autre en terme de répartition des week-ends, j'aurais préféré la solution me permettant d'être en week-end la k^ème^ semaine. »

lequel peut se traduire par :

$$\left\{\begin{array}{l} h \succ_{\mathrm{L}(wer),e} h' \;\Rightarrow\; h \succ_{(\mathrm{L}(wer),k),e} h' \\ \left.\begin{array}{l} \text{non}[h \succ_{\mathrm{L}(wer),e} h'] \text{ et} \\ \text{non}[h \prec_{\mathrm{L}(wer),e} h'] \text{ et} \\ \exists i \in \{1,\ldots,m(h)\} \text{ tel que}: r(h,i) = k \text{ et} \\ \forall i \in \{1,\ldots,m(h)\},\; r(h',i) \neq k \end{array}\right\} \Rightarrow h \succ_{(\mathrm{L}(wer),k),e} h' \end{array}\right.$$

avec $\succcurlyeq_{(\mathrm{L}(wer),\,k),\,e}$ les préférences de l'individu *e* concernant la répartition des week-ends complets au repos, et sa préférence pour que le week-end *k* soit compté plus que les autres, sans pour autant remettre en cause les préférences entre deux horaires n'ayant pas le même nombre de week-ends, ou n'ayant pas une répartition équivalente. Considérons par exemple l'horaire h_4 décrite en Figure 45 (*a*) avec $k = 6$.

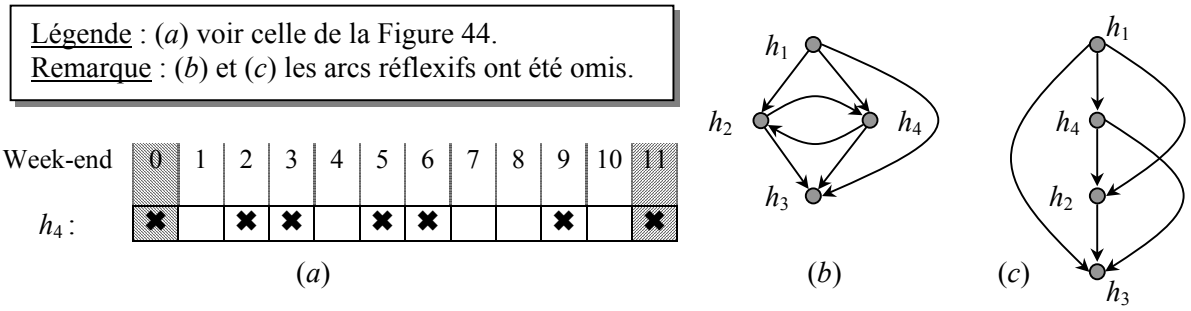


Figure 45 : Exemple de préférences individuelles : (*a*) Définition de l'horaire h_4 et, Comparaison avec les horaires h_1, h_2 et h_3, (*b*) selon la relation $\succcurlyeq_{\mathrm{L}(wer),\,e}$ et (*c*) selon la relation $\succcurlyeq_{(\mathrm{L}(wer),\,6),\,e}$.

Cet exemple montre une manière de gérer un conflit, en utilisant la récursivité : construire une relation binaire de préférence à partir d'une autre. Les contraintes souples présentées au § 5.2 sont particulièrement bien adaptées. Cette manière de procéder, via la récursivité, semble mieux adaptée à la gestion de conflits que l'intervention systématique du planificateur (gestion manuelle par des variations de poids), ou la délégation purement stratégique et arbitraire de la décision à l'algorithme.

6.4.2.3. Équité, Efficacité et Qualité de Vie Collective des Travailleurs

Procurer aux employés une qualité de vie pas seulement au travail mais aussi, autant que faire se peut, à l'extérieur, est un objectif qui devrait être visé par l'élaboration des plannings ; sachant que le temps consacré au travail est intimement lié aux temps sociaux (GAUVIN & JACOT [1999]). Cette qualité de vie passe bien sûr par une répartition avantageuse des horaires, mais aussi par une bonne hygiène et sécurité, une bonne qualité de vie en communauté, le droit à la parole, etc. au sein de l'organisation. Nous nous intéressons ici uniquement à la qualité de vie des employés engendrée par les plannings. Cette qualité de vie peut être décrite

de manière satisfaisante au moyen de règles d'agrégation véhiculant des principes d'équité et d'efficacité.

Il est souvent possible de fournir une liste des propriétés jugées pertinentes pour asseoir l'évaluation de la qualité de vie des travailleurs. Nous en fournissons une liste non exhaustive :

Quelques attributs de définition de l'espace d'évaluation :

- Les durées journalières et hebdomadaires d'un individu à l'autre ne doivent pas être trop différentes ; Par exemple, ce ne sont pas toujours les mêmes qui font des heures supplémentaires.
- le nombre de week-ends entièrement au repos doit être le plus grand possible et réparti au mieux entre les employés ;
- le nombre de week-ends partiellement au repos (samedi d'astreinte et dimanche au repos par exemple) doit être le plus petit possible et réparti au mieux entre les employés ;
- la proportion des différents types de vacations (matin, soir et nuit par exemple) doit être réparti équitablement entre les individus. Certains ne doivent pas faire plus de nuits que d'autres, alors qu'il n'est nullement spécifié que certaines personnes préfèrent travailler de nuit ;
- le nombre de changements de vacations doit être le plus petit possible et réparti au mieux entre les individus. Car cela brise les rythmes circadiens, favorisant le stress ;
- les séquences de vacations trop longues ou trop courtes doivent être minimisé pour chaque employé et distribué au mieux entre les employés ; Les séquences peuvent concerner un type de vacation particulier, ou l'ensemble des types de vacations (c'est-à-dire le nombre de jours de travail consécutifs) ;
- les repos isolés doivent être évités et partagés entre les employés ;
- les jours fériés doivent être équilibrés au mieux entre les individus, etc.

Chaque attribut est un vrai-critère étendu sur l'ensemble des positions (h, e) avec e un travailleur et h un horaire admissible pour cet employé. Le modèle préférentiel des PEPTP est caractérisé par un ensemble plutôt large d'attributs pertinents sur lequel repose l'évaluation des plannings. Quels sont les attributs pertinents pour une évaluation acceptable ? Tout élément ayant une importance dans la vie d'un employé (les moyens de fonctionnement, les capabilités[163]), ayant un temps soit peu une dimension temporelle (qui peut se traduire concrètement en plages horaires), ou influençable (pouvant être diminué ou valorisé) par l'agencement des horaires, peut être candidat. Ces attributs identifient l'espace d'évaluation.

Ces attributs peuvent être proposés par les employés. Il est aussi généralement acquis dans nos problèmes que des acteurs secondaires (syndicats, pouvoir législatif via le code du travail), donnent leur avis sur les conditions des travailleurs en identifiant un certain nombre d'attributs pertinents pour la comparaison des horaires de travail. Toutes ces informations préférentielles peuvent être utilisées pour concevoir une règle d'agrégation véhiculant l'équité et l'efficacité, et traduisant bien la notion de qualité de vie. Ces acteurs secondaires sont particulièrement importants lorsque le nombre d'employés n'est pas connu (PEPTREG et PEPTI).

La dominance de Lorenz généralisée multidimensionnelle (MGL) peut être utilisée pour agréger les différents attributs jugés pertinents de manière à évaluer la qualité de vie des travailleurs. Cette méthode, comme nous l'avons vu, pour qu'elle soit correctement utilisée doit être identifier un ensemble exhaustif d'attributs (§ 6.3.5.1). Cependant certains attributs ayant une importance négligeable pourront être ignorés. Les acteurs peuvent jouer un rôle actif dans cette identification. Pour améliorer l'évaluation, il est aussi possible de solliciter les employés pour qu'ils précisent certaines comparaisons interpersonnelles partielles (dans le même ordre d'idées du § 6.3.5.2).

[163] Voir par exemple SEN [1980], [1987*a*], [1987*b*], [1992], [1993*a*], FOSTER & SEN [1997].

Selon la classe du problème d'élaboration de plannings de travail, il est possible de compléter le modèle avec d'autres règles d'agrégation véhiculant l'équité. Ainsi, les PEPTA et PEPTAG pour lesquels le personnel peut émettre des préférences individuelles (souhaits, etc.), peuvent aussi user de la règle d'équité de Suppes-Sen (ou d'autres règles comme l'envie (§ 6.3.3.3)) pour agréger équitablement ces préférences généralement purement ordinales. Dans le cas des PEPTREG et PEPTI, le nombre d'employés est inconnu, et peut varier. La qualité de vie de ce personnel non permanent peut être évaluée en complétant la dominance de Lorenz généralisée multidimensionnelle d'une version multidimensionnelle de l'axiome d'invariance (§ 6.2.6.1) aux réplications de la population (MPRI). Le tableau suivant récapitule ces différentes utilisations.

	Type de PEPTP					
Éléments de base	PEPTA		PEPTAG		PEPTREG	PEPTI
	participatif	non participatif	participatif	non participatif		
MGL	✓	✓	✓	✓	✓	✓
SSE, ENVY	✓		✓			
MPRI					✓	✓

Tableau 7 : Récapitulatif des axiomes et règles d'agrégations véhiculant des principes d'équité, pouvant intervenir dans la définition de la qualité de vie de la population induite par un planning selon la classe du PEPTP. (Légende : une encoche (✓) signifie que la qualité de vie peut être définie à partir de cet élément d'agrégation**).**

Dans le cas des PEPTREG et PEPTI, l'axiome d'invariance MPRI et la dominance MGL sont complémetaires. Ils permettent de comparer partiellement mais équitablement et efficacement des plannings avec un nombre d'employés différent. Par contre, pour les PEPTA et PEPTAG participatifs, la définition d'une qualité de vie prenant en compte toutes les préférences exprimées, peut s'avérer être plus complexe. La règle d'équité de Suppes-Sen peut entrer en conflit avec la dominance de Lorenz multidimensionnelle. Le choix d'une règle d'agrégation finale est alors nécessaire.

Résolution Algorithmique :

Optimisation Combinatoire

Basée sur les Préférences

Au fil des études, les problèmes d'élaboration de plannings se sont avérés très difficiles (ardus) à comprendre, à modéliser mais aussi à résoudre.

Dans cette partie, nous argumentons sur le fait que la gestion des hommes, nécessitant une gestion particulière des préférences, elle requiert aussi des outils de résolution adaptés.

7. Algorithmes de Résolution de Problèmes d'Aide à la Décision Basés sur les Préférences et les Contraintes

Ce chapitre est consacré à la résolution de problèmes d'aide à la décision usant de relations binaires de préférences (§ 3.3.1). Ainsi, au § 7.1 des algorithmes de recherche arborescente sont présentés résolvant certains problèmes combinatoires d'aide au choix basés sur les préférences. Une parenthèse est ouverte sur la programmation multiobjective au § 7.2 pour présenter des formulations linéaires de la dominance de Lorenz et de sa version généralisée. Au § 7.3 nous présentons une manière de réduire le nombre de contraintes globales à concevoir pour une PB-contrainte souple. Le dernier paragraphe (§ 7.4) est consacré à la conception d'algorithmes de filtrage pour un ensemble de contraintes souples précédemment présentés : les dominances de Pareto et de Lorenz, ainsi que la règle de Pareto et la règle d'équité de Suppes-Sen. Dans ce cadre, nous voyons notamment comment des concepts de la théorie des graphes, comme les couplages parfaits et les couvertures de sommets, peuvent aider à concevoir des algorithmes de filtrage pour des contraintes souples.

7.1. Résolution de Problèmes de Choix Basés sur les Préférences et les Contraintes

Nous présentons maintenant des algorithmes résolvant certains problèmes de choix basés sur les préférences binaires ($\mathcal{R}$, $\{c_{\succcurlyeq}[\alpha, z]\}_{\alpha, z}$) ; Les trois premières sous-sections sont dédiées aux systèmes de contraintes basés sur les préordres partiels. La sous-section 7.1.4 effectue une ouverture sur les problèmes quasi-transitifs et acycliques.

Comment résoudre les PPO-CSP et les SM-PPO-CCP avec les ordinateurs ? De nombreux travaux ont effectué la recherche d'une solution maximale pour ($S_{\text{CSP}}(\mathcal{R})$, $\succcurlyeq$) au moyen d'une fonction objective temporaire (voir ROY & BOUYSSOU [1993, § 7.2.3]). Dans cette perspective, il est nécessaire d'identifier des fonctions objectives ayant leur ensemble optimal inclus dans $M(S_{\text{CSP}}(\mathcal{R}), \succcurlyeq)$ (ou $B(S_{\text{CSP}}(\mathcal{R}), \succcurlyeq)$). D'un autre coté, les contraintes globales donnent la possibilité de construire de complexes instances et de les résoudre ensuite, sans passer par une identification théorique de fonctions objectives. Cette possibilité de 'clientélisation' des modèles de préférence ouvre de grandes perspectives.

Ces dernières années, une nouvelle forme d'algorithmes type Branch & Bound[164] ont fait leur apparition pour résoudre ces problèmes d'aide au choix (GAVANELLI [2002*a*]). Après avoir décrit les deux algorithmes existants traitant de ce sujet en 7.1.1, nous en présentons un plus efficace au § 7.1.2 et 7.1.3.

7.1.1. Les Algorithmes de Gavanelli

GAVANELLI [2002] présente deux algorithmes résolvant le SM-PPO-CCP, lequel correspond à l'énumération d'un ensemble de choix maximisant stable maximal à partir d'un sys-

[164] Pour des références générales sur les algorithmes de type Branch & Bound et Backtrack, voir SAKAROVITCH [1984*b*], NEMHAUSER & WOLSEY [1988], THIENEL [1995], ZHANG [1996], FAGES [1996], FROST [1997], parmi d'autres.

tème de contraintes muni d'un préordre partiel. Ils sont décrits dans Algorithme 2 et Algorithme 3[165].

`Enum_PPO1`($\mathcal{K} = (V, D, C)$: système de contraintes, $\{c_{\succcurlyeq}[\alpha, y]\}_{\alpha, y}$: préordre partiel):
retourner {ensemble de choix maximisant stable maximal, non}
début
  $J = (V_J, D_J, C_J)$: système de contraintes $\leftarrow \mathcal{K}$ instance courante à résoudre.
  $x \in \mathcal{D}_V \cup \{\text{non}\} \leftarrow \text{non}$ solution courante initialisée à non = absence de solution réalisable
  $RM \in \mathcal{P}(\mathcal{D}_V) \leftarrow \varnothing$ ensemble de choix courant, initialisé à l'ensemble vide
  faire
    $x \leftarrow$ `Algo_CSP`(J) recherche d'une solution réalisable x
    si ($x \neq$ non) **alors** l'instance J est consistante
      $RM \leftarrow RM \cup \{x\} \setminus \{y \in RM$ tel que: $y \in S(c_{\succcurlyeq}[\text{pire}, x], V, D)\}$ la solution courante x est ajoutée aux solutions maximales potentielles, et celles qui sont pires que la solution courante sont supprimées
      $C_J \leftarrow C_J \cup \{c_{\succcurlyeq}[\{\text{mieux, incomparable}\}, x]\}$ les prochaines solutions réalisables recherchées doivent être mieux ou incomparables à x
    fin si
  tant que ($x \neq$ non)
  retourner RM
fin `Enum_PPO1`

Algorithme 2. Premier algorithme de GAVANELLI résolvant le SM-PPO-CCP en utilisant un oracle dédié au CSP.

Ce premier algorithme est une version d'algorithmes plus anciens de la littérature (voir GAVANELLI [2002] pour plus de détails). Comme son nom l'indique, `Algo_CSP`(J) est un algorithme exact résolvant le CSP en J, par exemple le backtracking ou tout autre algorithme dédié aux particularités de J (le simplexe ou les procédures de flots si J est une instance de ces problèmes).

`Enum_PPO2`($\mathcal{K} = (V, D, C)$: système de contraintes, $\{c_{\succcurlyeq}[\alpha, y]\}_{\alpha, y}$: préordre partiel):
retourner {ensemble de choix maximisant stable maximal, non}
début
  $\wp$: ensemble de systèmes de contraintes $\leftarrow \{\mathcal{K}\}$ ensemble des instances courantes à résoudre.
  $RM \in \mathcal{P}(\mathcal{D}_V) \leftarrow \varnothing$ ensemble courant de solutions maximales potentielles, initialisé à l'ensemble vide
  $x \in \mathcal{D}_V \cup \{\text{non}, \varepsilon_D\} \leftarrow \varepsilon_D$ solution associée à l'instance courante, initialisée à ε_D. non = absence de solution réalisable ; ε_D = consistance ou inconsistance de l'instance courante est inconnue.
  tant que ($\wp \neq \varnothing$) **faire**
    $J = (V_J, D_J, C_J) \leftarrow$ <u>choisir</u>($\wp$)
    $x \leftarrow$ <u>évaluer</u>(J) teste si une solution réalisable peut être obtenue à partir de l'instance J.
    si ($x =$ non) **alors** instance J est inconsistante.
      $\wp \leftarrow \wp \setminus \{J\}$
    sinon si ($x \notin \{\text{non}, \varepsilon_D\}$) **alors** x désigne une solution réalisable de J.
      pour tout ($J1 \in \wp$) **faire**
        la contrainte $c_{\succcurlyeq}[\{\text{mieux, incomparable}\}, x]$ est ajoutée à toutes les instances de $\wp$.
        $C_{J1} \leftarrow C_{J1} \cup \{c_{\succcurlyeq}[\{\text{mieux, incomparable}\}, x]\}$
      fin pour

---

[165] Remarque : Une erreur s'est glissée dans les algorithmes originaux de GAVANELLI, avec l'utilisation de la contrainte $c_{\succcurlyeq}[\{\text{surclasse, incomparable}\}, x]$ au lieu de $c_{\succcurlyeq}[\{\text{mieux, incomparable}\}, x]$. Cette première contrainte boucle à l'infini si la prochaine solution est x lui-même.

$RM \leftarrow RM \cup \{x\} \setminus \{y \in RM$ tel que: $y \in S(c_{\succcurlyeq}[\text{pire}, x], V, D)\}$ la solution courante x est ajoutée aux solutions maximales potentielles, et celles qui sont pires que la solution courante sont supprimées

sinon séparation (branch)

$\wp \leftarrow \wp \setminus \{J\} \cup$ <u>générer sous-instances</u>(J)

fin si

fin tant que

retourner RM

fin `Enum_PPO2`

Algorithme 3. **Second algorithme de GAVANELLI résolvant SM-PPO-CCP.**

Remarques 2 . *(sur `Enum_PPO2`)* Cet algorithme générique de type Branch-and-Bound peut ensuite être spécialisé en fonction des particularités de l'instance de CSP à résoudre, et de l'implémentation des stratégies de recherche : <u>choisir</u>($\wp$), <u>évaluer</u>(J) et <u>générer sous-instances</u>(J). Ces stratégies sont les mêmes que celles utilisées dans les algorithmes de Branch-and-Bound courants de la RO et de l'IA. Les **stratégies de choix** (implémentées dans <u>choisir</u>($\wp$)) choisissent une sous-instance dans $\wp$: recherche en profondeur ou en meilleur d'abord, etc. couplé à un retour arrière standard ou intelligent. Les **stratégies d'évaluation** (implémentées dans <u>évaluer</u>(J)) testent la consistance du système de contraintes J courant, établissent quelques fois un niveau de consistance sur J, et si possible, retournent une solution réalisable de J pouvant avoir certaines particularités, comme celle d'être optimale sur ($S_{\text{CSP}}(J)$, f_J), avec f_J une fonction objective temporaire (cas des algorithmes de Branch-and-Bound usuels). Les **stratégies de séparation** (implémentées dans <u>générer sous-instances</u>(J)) divisent l'espace de recherche associé à J en morceaux plus petits, par exemple via instanciation statique ou dynamique des variables.

La terminaison de `Enum_PPO2` dépend des stratégies utilisées. Les stratégies usuelles de séparation, de choix et d'évaluation aboutissent à des algorithmes finis.

Ces deux algorithmes permettent d'énumérer les éléments d'un ensemble de choix maximisant stable maximal, si un tel ensemble existe. Lorsque cet ensemble de choix est réduit à un seul élément, il est aussi optimisant (§ 3.4.2). Ces algorithmes s'appuient sur la contrainte souple $c_{\succcurlyeq}[\{\text{mieux}, \text{incomparable}\}, x]$, et sur le problème de comparaison ($y \in S(c_{\succcurlyeq}[\text{pire}, x], V, D)$?).

La différence entre ces deux algorithmes est que l'algorithme `Enum_PPO1` recherche une solution réalisable sans tenir compte de l'arbre de recherche ayant servi à trouver la solution réalisable précédente, contrairement à `Enum_PPO2`.

Les deux algorithmes de GAVANELLI ont deux principaux inconvénients. Le certificat prouvant qu'une solution trouvée, réalisable pour CSP($\mathcal{R}$), est maximale, est fourni en même temps que le certificat prouvant que l'ensemble de solutions potentielles est aussi un ensemble de choix maximisant (ou optimisant) stable maximal. Et, dans le pire des cas, l'information mémorisée (nombre de contraintes ajoutées) peut être exponentielle en la taille à la fois de l'entrée (le système de contraintes basé sur les préférences) et la sortie (un ensemble de choix maximisant stable maximal). GAVANELLI l'a d'ailleurs signalé dans sa thèse [2002*a*]. Ces inconvénients limitent leur utilisation pratique à de très petits problèmes (comme le faisait remarquer TORRENS & FALTINGS [2002]).

Un algorithme sortie-polynomial en espace, résolvant le problème d'énumération de toutes les solutions d'un ensemble maximisant stable maximal, est décrit ci-dessous. Cet algorithme est formé de deux parties. La première cherche une solution maximale (§ 7.1.2) en utilisant la contrainte globale $c_{\succcurlyeq}[\text{mieux}, x]$, et le second limite l'espace de recherche aux solutions maximales incomparables non encore trouvées (§ 7.1.3) avec la contrainte $c_{\succcurlyeq}[\text{incomparable}, x]$.

7.1.2. *Trouver une Solution Maximale par Branch and Bound*

Nous présentons ici un algorithme résolvant partiellement PPO-CSP, en cherchant une solution maximale, si une telle solution existe, sinon retournant "non", à partir d'un système de contraintes basé sur un préordre partiel. Cet algorithme de type Branch-and-Bound est décrit par l'Algorithme 4.

`B&B_PPO`($\mathcal{K} = (V, D, C)$: système de contraintes, $\{c_{\succcurlyeq}[\alpha, y]\}_{\alpha, y}$: préordre partiel) :
retourner { solution maximale, non}
début
  $\wp$: ensemble de systèmes de contraintes $\leftarrow \{\mathcal{K}\}$ ensemble des instances courantes à résoudre.
  $x \in \mathcal{D}_V \cup \{\text{non}, \varepsilon_D\} \leftarrow \varepsilon_D$ (même sens que pour `Enum_PPO2`)
  $x_1 \in \mathcal{D}_V \cup \{\text{non}\} \leftarrow$ non meilleure solution courante, initialisée à non $\Leftrightarrow$ aucune solution réalisable n'a encore été trouvée pour le moment.
  tant que ($\wp \neq \varnothing$) **faire**
    $J = (V_J, D_J, C_J) \leftarrow$ <u>choisir</u>($\wp$)
    $x \leftarrow$ <u>évaluer</u>(J) teste si une solution réalisable peut être obtenue à partir de l'instance J.
    si (x = non) **alors** l'instance J est inconsistante.
      $\wp \leftarrow \wp \setminus \{J\}$
    sinon si ($x \notin \{\text{non}, \varepsilon_D\}$) **alors** x désigne une solution réalisable pour J.
      pour tout ($J1 \in \wp$) **faire** la contrainte $c_{\succcurlyeq}$[mieux, x] est ajoutée à toutes les instances de $\wp$.
        $C_{J1} \leftarrow C_{J1} \cup \{c_{\succcurlyeq}[\text{mieux}, x]\}$
        si ($x_1 \neq$ non) **alors** $C_{J1} \leftarrow C_{J1} \setminus \{c_{\succcurlyeq}[\text{mieux}, x_1]\}$ **fin si**
          supprime la contrainte redondante $c_{\succcurlyeq}[\text{mieux}, x_1]$
        $x_1 \leftarrow x$
      fin pour
    sinon séparation (branch)
      $\wp \leftarrow \wp \setminus \{J\} \cup$ <u>générer sous-instances</u>(J)
    fin si
  fin tant que
  retourner x_1 ici, soit x_1 est une solution réalisable et maximale pour $\succcurlyeq$ dans $\mathcal{K}$, soit x_1 = non.
fin `B&B_PPO`

Algorithme 4. **Un algorithme de type Branch and Bound pour chercher une solution maximale pour un préordre partiel parmi les solutions réalisables d'un système de contraintes.**

Les Remarques 2 sont aussi valables pour `B&B_PPO`. Comme `Enum_PPO2`, la terminaison de `B&B_PPO` dépend des stratégies de séparation, choix et évaluation utilisées.

Supposons maintenant que les stratégies utilisées produisent un algorithme fini pour `B&B_PPO`. Alors cet algorithme peut trouver n'importe quelle solution maximale d'un système de contraintes donné selon un préordre partiel donné (admissibilité), si une telle solution existe et si la stratégie <u>évaluer</u>(J) est non biaisée. La preuve est directe grâce aux propriétés des préordres partiels (Propriétés 4 page 153). En conséquence, les instances inconsistantes courantes, qui sont ensuite supprimées pour réduire l'espace de recherche, ne contiennent aucune solution maximale. Par exemple, si <u>évaluer</u>(J) est un algorithme de filtrage alors l'admissibilité est prouvée.

7.1.3. *Énumérer Tous les Éléments d'un Ensemble de Choix Maximisant Stable Maximal*

La figure suivante propose de résoudre SM-PPO-CCP par un algorithme incrémental polynomial en espace par rapport aux tailles de l'entrée et de la sortie, tout en améliorant les performances dans le pire cas des algorithmes précédents.

```
Enum_PPO3(ℛ = (V, D, C): système de contraintes, {c≽[α, y]}α, y: préordre partiel):
                                    retourner {ensemble de choix maximisant stable maximal, non}
début
   J = (VJ, DJ, CJ) : système de contraintes ← ℛ          instance courante à résoudre.
   x ∈ 𝒟V ∪ {non} ← non      solution courante initialisée à non = absence de solution
   RM ∈ 𝒫(𝒟V) ← ∅            ensemble de choix courant, initialisé à l'ensemble vide
   faire
      x ← B&B_PPO(J, {c≽[α, y]}α, y)        recherche d'une solution maximale x
      si ( x ≠ non ) alors        l'instance J est consistante
         RM ← RM ∪ {x}        la solution courante x est une nouvelle solution maximale
         CJ ← CJ ∪ {c≽[incomparable, x]}      solutions réalisables incomparables à x
      fin si
   tant que ( x ≠ non )
   retourner RM
fin Enum_PPO3
```

Algorithme 5. **Un algorithme incrémental retournant un ensemble de choix maximisant stable maximal pour un préordre partiel sur un système de contraintes.**

Cet algorithme est complet, i.e. il retourne un ensemble de choix maximisant stable maximal tout entier, si B&B_PPO est admissible. Il est fini si B&B_PPO est fini et $\mathcal{R}$ est un système de contraintes fini.

Par rapport aux algorithmes de GAVANELLI, B&B_PPO a une complexité en espace qui reste polynomiale en la taille de l'entrée et de la sortie (une contrainte est ajoutée dès qu'une solution maximale est trouvée). L'espace de recherche est élagué plus efficacement grâce à l'utilisation séparée des attitudes 'mieux' et 'incomparable'. Cette utilisation séparée permet de diminuer de manière plus importante, le nombre de solutions réalisables, tout en laissant intactes les solutions maximales. De plus, les solutions maximales sont trouvées itérativement, à cause de l'utilisation de la contrainte $c_{\succcurlyeq}$[mieux, x] au lieu de $c_{\succcurlyeq}$[{mieux, incomparable}, x]. Cette première contrainte a un plus fort pouvoir d'élagage. Remarquons qu'à partir du moment où une solution réalisable est trouvée, une solution est disponible à tout moment et sa qualité selon la relation de préférence croit avec le temps.

La terminaison de cet algorithme n'est pas une conséquence directe du fait que l'ensemble des solutions réalisables est fini (problème combinatoire) : la possibilité de générer des sous-instances d'instances dont l'ensemble des solutions réalisables est vide, et celle de ne pas toujours pouvoir affirmer qu'une instance donnée est sans solution réalisable, interdit en effet de conclure dans le cas général.

Remarquons que lorsque le nombre de solutions réalisables de $\mathcal{R}$ est infini, il est possible d'utiliser cet algorithme, mais en utilisant des stratégies d'évaluation et de séparation adéquates (voir par exemple la résolution des MILPs en RO). En d'autres termes, lorsque $S_{\text{CSP}}(J)$ n'est pas fini, on peut se limiter aux conditions précédentes quand la hauteur[166] de $(\mathcal{D}_V, \succcurlyeq)$ est polynomiale en la taille de l'instance. Mais généralement (lorsque la hauteur est exponentielle, voire indénombrable ; mais aussi pour de meilleures performances algorithmiques), il est nécessaire de trouver une solution réalisable x vérifiant : $S_{\text{CSP}}(V_J, D_J, C_J \cup \{c_{\succcurlyeq}[\text{mieux}, x]\}) = \emptyset$. La consistance qualitative (§ 5.3.5) n'est pas loin !

Ces algorithmes généraux sont utiles que s'ils sont plus performants qu'un autre effectuant la même tâche, mais aussi lorsqu'aucune extension en vrai-critère, du préordre partiel $\succcurlyeq$, n'est connue. Ce qui est le cas lorsque le modèle de préférence est complexe et multi-niveaux

[166] La hauteur d'un ensemble partiellement préordonné $(S, \succcurlyeq)$ est le plus grand cardinal d'un sous-ensemble de S tel que $(B, \succcurlyeq)$ est un ensemble totalement ordonné (voir TROTTER [1992]).

d'agrégation. De plus, ces algorithmes généraux sont un bon point de départ pour élaborer par la suite des méthodes hybrides, mixtes exactes (algorithmes efficaces de RO) et approchées (amélioration locale, méta-heuristiques).

Remarquons aussi que ces algorithmes peuvent être adaptés au cas où une solution réalisable est connue, obtenue par exemple par une heuristique. Ces algorithmes servent de base à plusieurs problèmes pouvant se ramener à un POP-CSP (comme les problèmes sur-contraints). L'algorithme `B&B_POP` est utilisé comme un générateur de solutions maximales, et à ce titre peut aussi être utilisé pour résoudre des problèmes d'énumération partielle. Par exemple énumérer au plus $k > 0$ solutions maximales[167] d'un système de contraintes basé sur les préférences. Une autre utilité de taille, nous l'avons vu, concerne la résolution interactive de problèmes dont l'ensemble maximal est trop volumineux, comme c'est souvent le cas.

L'algorithme d'énumération `Enum_PPO3` ne tient aucun compte de l'information accumulée durant la recherche des solutions maximales précédentes. Certaines branches de l'arbre de recherche ne sont plus à parcourir, en plus des branches supprimées par l'ajout de la contrainte d'incomparabilité[168]. Il y a ainsi une perte par rapport au second algorithme de GAVANELLI.

Spécialisation et gain en performance algorithmique : Ces algorithmes généraux ne tiennent pas compte de connaissances disponibles sur le préordre partiel : bornes, nombre de classes d'équivalence, hauteur du préordre, solution idéale, extension en vrai-critère, etc.. Ces connaissances peuvent être utilisées pour améliorer les performances en spécialisant les algorithmes.

7.1.4. Au-Delà des Préordres Partiels

Peu de résultats sont disponibles sur la résolution de problèmes combinatoires évalués par une relation binaire de préférence possédant des propriétés structurales plus faibles que le préordre partiel. Notons néanmoins les travaux très récents de PERNY & SPANJAARD [2002] sur les graphes d'états. Ils proposent un algorithme résolvant les problèmes de choix sur les graphes d'espace d'états évalués par une relation binaire de préférence quasi-transitive.

L'algorithme `B&B_PPO` (Algorithme 4) peut être utilisé pour trouver une solution maximale d'un système de contraintes fini basé sur une relation binaire de préférence quasi-transitive (résolvant partiellement le QT-CSP). En effet, il utilise seulement la contrainte $c_{\succcurlyeq}[\text{mieux}, x]$ et toute relation binaire de préférence quasi-transitive vérifie une monotonie de l'inclusion des solutions réalisables par rapport à la préférence stricte de la relation binaire de préférence (relation (c) des Propriétés 4 en page 153), propriété nécessaire pour réduire l'espace de recherche. De plus, il peut atteindre (retourner) n'importe quelle solution maximale (admissibilité). Par contre, s'il existe une solution optimale, il n'en retourne pas forcément une.

L'algorithme `Enum_PPO3` ne peut pas être utilisé pour trouver tout un ensemble maximisant stable maximal de $S_{\text{CSP}}(\mathcal{P})$ selon une relation binaire de préférence quasi-transitive $\succcurlyeq$ quelconque modélisée par $\{c_{\succcurlyeq}[\alpha, y]\}_{\alpha, y}$. Illustrons cette affirmation avec l'exemple où $S_{\text{CSP}}(\mathcal{P}) = \{x, y, z\}$ et $x \simeq y, y \succ z, x \parallel z$. La relation $\succcurlyeq$ est quasi-transitive et $M(S_{\text{CSP}}(\mathcal{P}), \succcurlyeq) = \{x, y\}$. Par conséquent, $SM(S_{\text{CSP}}(\mathcal{P}), \succcurlyeq) \in \{ \{y\}, \{x\} \}$. L'algorithme `Enum_PPO3` peut par contre retourner $\{y\}$ ou $\{x, z\}$, bien que $\{x, z\}$ ne soit pas un sous-ensemble maximal. Si l'ensemble retourné est formé d'un seul élément, alors cet ensemble est maximisant. Mais il n'est pas forcément optimisant (§ 3.4.2). De plus, cet algorithme ne retourne pas systématiquement un ensemble de choix optimisant stable maximal quand $B(S_{\text{CSP}}(\mathcal{P}), \succcurlyeq) \neq \varnothing$. Prenons l'exemple de SEN [1997] où $S_{\text{CSP}}(\mathcal{P}) = \{x, y, z\}$ et $x \simeq y, y \simeq z, x \parallel z$. La relation $\succcurlyeq$ est quasi-transitive et

[167] Pour une illustration de ce problème, voir TORRENS & FALTINGS [2002].

[168] L'apprentissage est une autre argumentation à l'utilité de considérer le processus dans sa globalité, et non pas uniquement comme une séquence de problèmes partiels (§ 3.2.2.6).

$\{y\} = B(S_{CSP}(\mathcal{K}), \succcurlyeq) \neq M(S_{CSP}(\mathcal{K}), \succcurlyeq) = \{x, y, z\}$. Par conséquent, $SB(S_{CSP}(\mathcal{K}), \succcurlyeq) \in \{ \{y\} \}$ et $SM(S_{CSP}(\mathcal{K}), \succcurlyeq) \in \{ \{y\}, \{x, z\} \}$. L'algorithme `Enum_PPO3` peut alors retourner $\{y\}$ ou $\{x, z\}$ mais pas toujours $\{y\}$.

Dans le cas de la relation binaire acyclique, `B&B_PPO` n'est pas adapté, parce que la relation (c) des Propriétés 4 n'est pas vérifiée.

7.2. Programmation MultiObjective et Dominance de Lorenz

Faisons un petit détour vers la programmation multiobjective (MOP). Au § 6.4, nous avons décidé de fonder le modèle de préférence de nos problèmes d'élaboration de plannings de personnel sur deux concepts que sont l'efficacité et l'équité. L'efficacité, modélisée par la dominance et la règle de Pareto, a été beaucoup étudiée en RO et en TCS (voir EHRGOTT & GANDIBLEUX [2000] et § 4.2.1.3). Concernant l'équité, la dominance de Lorenz et ses extensions ainsi que la règle de Suppes-Sen ont été presqu'exclusivement étudiées en TCS. Nous nous focalisons ici sur la dominance de Lorenz L et sa variante généralisée GL.

Pour le moment, très peu d'applications ont été modélisées en programmes mathématiques multiobjectifs basés sur ces dominances. Bien que l'utilisation de fonctions objectives temporaires ne soit pas obligatoire pour résoudre un programme mathématique multiobjectif (§ 2.1.3.2), elles forment une réelle alternative pour l'implémentation de stratégies d'évaluation (évaluer(J) au § 7.1.2) très performantes.

La programmation par buts peut-elle servir à rechercher des solutions maximales en compréhension pour la dominance de Lorenz ? Le § 7.2.1 effectue une étude partielle des propriétés axiomatiques de programmes par buts. Ce qui permettra de mieux comprendre leur fonctionnement. Le § 7.2.2 se tourne ensuite vers la transformation des MOPs agrégés par la dominance de Lorenz en MOPs agrégés par la dominance de Pareto ayant même ensemble maximal. En complément des travaux de KOSTREVA & OGRYCZAK [1999], nous montrons comment le multipermutoèdre peut être utilisé dans ce cadre.

7.2.1. Programmation par Buts et Dominance de Lorenz

A la section 6.2.2.1 nous n'avions pas envisagé l'utilisation de variables d'écart pour concevoir un modèle linéaire dont la fonction objective véhicule certains principes d'équité (§ 6.2.1). Cette sous-section montre que la programmation par buts définit des modèles vérifiant un ensemble d'axiomes d'équité.

7.2.1.1. Rappels sur la Programmation par Buts

La **programmation par buts** (*goal programming*), branche de l'optimisation multiobjective (Charnes & Cooper [1977]), a été conçue exactement pour modéliser les problèmes où l'objectif est de s'approcher au mieux d'une valeur cible fixée au départ.

L'instance d'un programme linéaire en nombre mixtes par buts, noté I_{MILGP}, est formée de :

- Un système d'inégalités linéaires en nombre mixtes décrit par le couple : $((A_1\ A_2), b)$, avec b un q-vecteur rationnel, A_1 et A_2 des matrices rationnelles de dimension respective $q \times m_1$ et $q \times m_2$.
  Le domaine réalisable est alors entièrement défini :
  $$S(I_{MILGP}) = \{x = (x_1, x_2) \in \mathbb{Q}^{m_1} \times \mathbb{Z}^{m_2} \text{ tel que : } A_1x_1 + A_2x_2 \geq b\}$$
- Un profil de vrai-critères linéaires : $\mathcal{F}(E, S(I_{MILGP})) = (f_1, f_2)$, avec f_1 et f_2 des matrices rationnelles de dimension respective $n \times m_1$ et $n \times m_2$, l'entier n étant le nombre de critères : $n = \text{card}(E)$;

– Une valeur cible (ou but) par critère : ob^* un n-vecteur rationnel fini[169].

La méthodologie consiste à se ramener à un programme multiobjectif linéaire en nombres mixtes de la manière suivante : A chaque vrai-critère du profil est associé deux paramètres appelés respectivement déviation positive d^+ et négative d^-. Puis une contrainte d'égalité est ajoutée, formée de la somme du vrai-critère et de la déviation négative auquel est soustrait la déviation positive au premier membre, et du but à atteindre pour ce critère au second membre. L'objectif est alors de minimiser le vecteur des déviations :

Minimiser $(d^+(1), \ldots, d^+(n), d^-(1), \ldots, d^-(n))$

$$\text{sous les contraintes : } \begin{cases} f_1(i,.)\cdot x_1 + f_2(i,.)\cdot x_2 + d^-(i) - d^+(i) = ob^*(i) \quad \text{pour tout } i \in E \\ A_1x_1 + A_2x_2 \geq b \\ d^+(i) \geq 0 \quad \text{et} \quad d^-(i) \geq 0 \quad \text{pour tout } i \in I. \end{cases} \tag{7–34}$$

Le vecteur des déviations forme un profil de $2n$ vrai-critères qui peut être ensuite agrégé (chap. 4). La règle d'agrégation usuellement utilisée est la somme pondérée suivante : $\sum_{i\in E} w(i)\cdot\left(d^+(i)+d^-(i)\right)$. Une solution optimale (x_1, x_2, d^+, d^-) du problème monocritère obtenu identifie une solution réalisable (x_1, x_2) de I_{MILGP} appelée solution optimale du problème d'optimisation par buts. Dans le cas de ressources multidimensionnelles (ressources formées d'attributs incommensurables), une hiérarchie agrégative peut être utilisée, comme nous allons le voir au § 7.2.1.3.

7.2.1.2. Programmation par Buts pour le Problème de Pure Répartition

Appliquons la programmation par buts au problème de pure répartition (E, Q). En prenant comme critère d'agrégation des déviations, la somme simple (la pondération forme un vecteur dont chaque composante vaut 1). Nous obtenons le programme linéaire suivant :

$$\text{Minimiser } \sum_{i\in E} d^+(i) + d^-(i)$$

$$\text{sous les contraintes : } \begin{cases} x(i) + d^-(i) - d^+(i) = x^*(i) \quad \text{pour tout } i \in E \\ \sum_{i\in E} x(i) = Q \\ x(i) \geq 0, \quad d^+(i) \geq 0 \quad \text{et} \quad d^-(i) \geq 0 \quad \text{pour tout } i \in E. \end{cases} \tag{7–35}$$

Remarquons que le score du programme linéaire (7–35) en une solution x de $S(I_{PR})$ donnée, est aussi obtenue en faisant la somme sur $i \in E$ des valeurs absolues $| x^*(i) - x(i) |$. Le programme linéaire (7–35) est donc une « linéarisation » du programme mathématique minimisant la distance $VA(x) = \sum_{i\in E}\left|x^*(i) - x(i)\right|$ sur l'ensemble des solutions réalisables[170] $S(I_{PR})$. Par conséquent, pour une solution z de $S(I_{PR})$ fixée, les déviations $d^+(i)$ et $d^-(i)$ étant variables, la valeur minimale du LP (7–35) est égale à $VA(z)$.

Nous avons alors la proposition suivante :

Proposition 12 : Le critère VA satisfait l'axiome d'égalité des proportions (EP) et l'invariance sous les permutations (IP). Par contre, VA n'est pas une extension du principe de transfert de Pigou-Dalton (PDT) pour card(E) > 2, ni de la règle de Lorenz pour card(E)

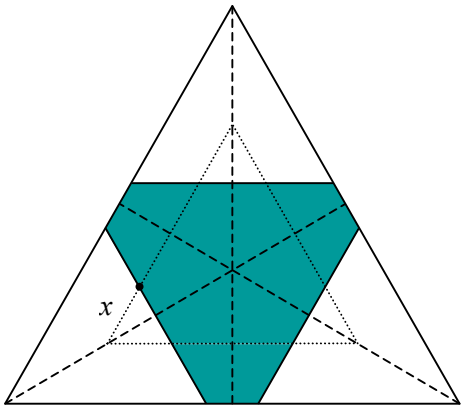
[169] Nous présentons ici uniquement le cas où les valeurs cibles sont finies. Dans le cas d'une valeur cible infinie, des adaptations sont nécessaires (voir référence précitée).

[170] Cette distance n'est pas une extension de la dominance de Lorenz ; contrairement à la distance euclidienne, correspondant au coefficient de variation à un facteur multiplicatif strictement positif près (voir note de bas de page numéro 134 en page 172).

> 2. En d'autres termes, pour x et y deux solutions de $S(I_{PR})$, si $y \succ_L x$ alors l'inégalité stricte $VA(x) > VA(y)$ n'est pas vérifiée.

Preuve : Voir Annexe B3.

La Figure 46 montre sur le triangle de Kolm le découpage du domaine réalisable obtenu à partir d'une solution x. La règle de Lorenz n'est pas vérifiée uniquement sur la frontière de la partie centrale. En dehors des solutions jugées équivalentes à x selon Lorenz (solutions obtenues par permutation), il y a incompatibilité.



La partie centrale de couleur foncée représente l'ensemble des solutions dont le score est plus petit que celui de la solution x. La frontière est l'ensemble des solutions équivalentes, et le reste du triangle de Kolm (les trois triangles de couleur blanche) identifie les solutions ayant un score plus élevé.

Figure 46 : Découpe du domaine réalisable du problème de pure répartition effectuée par la distance *VA* en *x*.

La proposition suivante complète la comparaison entre la dominance de Lorenz et la distance de la somme des valeurs absolues VA :

Proposition 13 : Pour tout x, y de $S(I_{PR})$, si $x \succcurlyeq_L y$ alors $VA(x) \leq VA(y)$. La réciproque est fausse.

Preuve : Voir Annexe B3.

Ainsi, la somme simple des déviations (vrai-critère de (7–35)) – ou encore son équivalent non linéaire VA – caractérise une règle d'agrégation qui ne satisfait pas le principe de transfert de Pigou-Dalton. Ce vrai-critère n'est pas une extension de la règle de Lorenz $\succcurlyeq_L$ (Proposition 12). Par contre, la règle de Lorenz y est incluse (Proposition 13). Quelles sont les conséquences de ces deux propositions sur les ensembles de choix ?

Nous allons poursuivre l'analyse des liens entre ces deux règles d'agrégation, en comparant leur ensemble de choix $C(S, \succcurlyeq_L)$ et $C(S, (VA, \leq))$ pour les sous-ensembles non vides S de $S(I_{PR})$. Remarquons que $C(S, \succcurlyeq_L)$ est un ensemble maximal $C(S, \succcurlyeq_L) = M(S, \succcurlyeq_L)$ dans le cas général, car la règle de Lorenz est un préordre partiel. $C(S, (VA, \leq))$ est par contre un ensemble optimal $C(S, (VA, \leq)) = B(S, (VA, \leq))$ car VA est un vrai-critère. Nous chercherons à savoir s'il existe des ensembles S pour lesquels des solutions de $B(S, (VA, \leq))$ ne sont pas dans $M(S, \succcurlyeq_L)$ et inversement. Et nous montrerons que l'intersection entre ces deux ensembles n'est jamais vide sauf si S est vide ou infini.

Proposition 14 : Étant donnée une instance de problème de pure répartition I_{PR}, les assertions suivantes sont vraies :

1) Il existe des ensembles non vides $S \subseteq S(I_{PR})$ tels que : $B(S, (VA, \leq)) \setminus M(S, \succcurlyeq_L) \neq \varnothing$;
2) Il existe des ensembles non vides $S \subseteq S(I_{PR})$ tels que : $M(S, \succcurlyeq_L) \setminus B(S, (VA, \leq)) \neq \varnothing$;
3) Pour tout sous-ensemble fini non vide S de $S(I_{PR})$, nous avons :

$$M(S, \succcurlyeq_L) \cap B(S, (VA, \leq)) \neq \varnothing.$$

Preuve : Voir Annexe B3.

L'assertion 3 nous assure que si l'on utilise la règle d'agrégation $(VA, \leq)$ pour chercher la solution la plus équitable d'un ensemble fini S inclus dans $S(I_{PR})$, alors au moins une de ces

solutions optimales est aussi maximale pour la dominance de Lorenz. Il serait maintenant intéressant de savoir si pour tout ensemble S inclus dans $S(I_{PR})$, $B(S, (VA, \leq)) \subseteq M(S, \succcurlyeq_L)$. Dans ce cas toute solution optimale de S serait Lorenz-efficace. Les assertions 1 et 2 prouvent le contraire.

Bien d'autres questions sont encore sans réponse concernant la comparaison de ces deux ensembles de choix, notamment les caractéristiques des ensembles S égalisant les deux ensembles ($\Leftrightarrow M(S, \succcurlyeq_L) = B(S, (VA, \leq))$) ou impliquant l'inclusion dans un sens ou dans un autre ($\Leftrightarrow B(S, (VA, \leq)) \subseteq M(S, \succcurlyeq_L)$ ou $B(S, (VA, \leq)) \supseteq M(S, \succcurlyeq_L)$).
De plus, est-il possible d'utiliser d'autres poids de manière à obtenir de meilleurs résultats ?
Nous laissons le reste de cette analyse axiomatique pour de futurs travaux. Notre but déclaré (c.f. fin de § 6.1) est d'identifier des modèles qui nous permettent d'avoir accès si nécessaire à toute solution de l'ensemble de choix collectif de notre problème, et non pas uniquement à un sous-ensemble très limité. Par conséquent, il est important de connaître les systèmes de poids aboutissant à des règles collectives possédant la propriété d'identifier des solutions optimales contenues dans l'ensemble de choix de notre problème. Les degrés de liberté offerts par ces poids permettront d'utiliser ces modèles préférentiels comme fonction d'agrégation temporaires (c.f. ROY & BOUYSSOU [1993, § 7.2.3]).
Ainsi, plus généralement, quelles sont les instances (S, RA), avec RA une somme pondérée des déviations, garantissant l'égalité ou l'inclusion des ensembles ?

7.2.1.3. Programmation par Buts et Pure Répartition Multidimensionnelle

Intéressons-nous maintenant succinctement à ce qui se passe dans le cas de plusieurs ressources multi-attribut. Le programme par buts direct associé au problème de pure répartition multidimensionnelle (§ 6.2.7) est le suivant :

Minimiser $d^+(r, i) + d^-(r, i)$ pour tout $i \in E$, et pour tout $r \in R$,

$$\text{Sous les contraintes : } \begin{cases} x(r,i) + d^-(r,i) - d^+(r,i) = x^*(r,i) \quad \text{pour tout } (r,i) \in R \times E \\ \sum_{i \in E} x(r,i) = Q(r) \\ x(r,i) \geq 0, \quad d^+(r,i) \geq 0 \quad \text{et} \quad d^-(r,i) \geq 0 \quad \text{pour tout } (r,i) \in R \times E. \end{cases}$$

La famille de $\mathrm{card}(R) \times \mathrm{card}(E)$ critères à minimiser est généralement agrégée via la dominance de Pareto ou une somme pondérée. Vérifient-ils les axiomes associés à une répartition multidimensionnelle équitable ? L'axiome de Lorenz uniforme (UL) n'est pas respecté si nous utilisons une somme pondérée, et que nous ayons un seul attribut ($\mathrm{card}(R) = 1$) comme nous venons de la voir. Nous nous limitons ici à comparer sur un exemple les règles d'agrégation précitées, à l'axiome de transfert corrélation croissante (CIT).

$$\text{Exemple : } x = \begin{pmatrix} 1 & 2 & 3 \\ 2 & 3 & 1 \\ 3 & 2 & 1 \end{pmatrix}, y = \begin{pmatrix} 1 & 2 & 3 \\ 1 & 3 & 2 \\ 1 & 2 & 3 \end{pmatrix} \text{ et } z = \begin{pmatrix} 1 & 2 & 3 \\ 1 & 2 & 3 \\ 1 & 2 & 3 \end{pmatrix}$$

La solution y est dérivée de x par une opération de transfert corrélation croissante entre la première et la troisième personne ; il en est de même pour z par rapport à y. En définitive, si l'on note $\succcurlyeq_{CIT}$ la règle d'agrégation associée à l'axiome CIT, nous avons : $x \succ_{CIT} y \succ_{CIT} z$. Le score des $\mathrm{card}(R) \times \mathrm{card}(E)$ critères sur les solutions x, y, z sont respectivement les matrices :

$$(d_x^+ + d_x^-) = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \end{pmatrix}, (d_y^+ + d_y^-) = \begin{pmatrix} 1 & 0 & 1 \\ 1 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix} \text{ et } (d_z^+ + d_z^-) = \begin{pmatrix} 1 & 0 & 1 \\ 1 & 0 & 1 \\ 1 & 0 & 1 \end{pmatrix}$$

- la règle de Pareto appliquée sur ces matrices critères informe que les solutions sont incomparables entre elles ;
- la somme simple sur E des matrices critères invite à comparer les vecteurs critères lignes : (2 1 3), (3 1 2) et (3 0 3).
  La règle de Pareto appliquée sur ces vecteurs critères lignes indiquent que les solutions sont toutes incomparables.
- la somme simple sur R des matrices critères invite à comparer les vecteurs critères colonne : $(2\ 2\ 2)^T$, $(2\ 2\ 2)^T$ et $(2\ 2\ 2)^T$. Ces solutions sont équivalentes, quelque soit la règle d'agrégation appliquée aboutissant à une relation binaire de préférence (propriété de réflexivité).

Les sommes simples et la règle de Pareto ne conduisent à aucune règle d'agrégation satisfaisant l'axiome CIT. D'autres possibilités sont offertes par les sommes pondérées. Les poids peuvent ou non dépendre des deux paramètres r et i, ou de l'un d'entre eux. Le but est alors d'étudier s'il existe des poids aboutissant à la satisfaction de l'axiome CIT. Dans le cas d'une réponse positive, pour quels systèmes de poids l'axiome CIT est-il satisfait ? Nous nous arrêtons ici dans la comparaison de ces règles d'agrégation.

En conclusion, l'analyse axiomatique, que nous avons partiellement effectuée pour des modèles de la programmation par buts, doit précéder toute adoption de modèle linéaire, pour identifier des critères de plus grande qualité. Par conséquent, une plus grande concentration des efforts de modélisation sur les axiomes, permettrait d'obtenir des modèles plus fiables, de meilleure qualité, plus robustes et plus aptes à assister le décideur.

7.2.2. Polyèdre Convexe et Dominance de Lorenz.

7.2.2.1. Multipermutoèdre et MultiObjectif Agrégé par la Dominance de Lorenz

Certaines règles d'agrégation RA de vrai-critères individuels possèdent des propriétés géométriques qui peuvent être avantageusement exploitées, pour transformer des programmes multiobjectifs usant de RA en programmes multiobjectifs équivalents pour l'ensemble des solutions maximales. C'est le cas de la dominance de Lorenz et de sa version généralisée. Étudions le problème multiobjectif suivant :

GL-MOP : Étant donnés un système d'inégalités en nombres mixtes $MIS \subseteq {}^m$, un ensemble d'individus observés I_e et un vrai-critère étendu $g = (f, \geq)$ de ${}^m \times I_e$ dans (⇔ un profil de $\mathrm{card}(I_e)$ vrai-critères simples de ^{m} dans dont les valeurs sont comparables, comme il est généralement défini), trouver une solution réalisable x qui maximise la dominance de Lorenz généralisée du vrai-critère étendu sur MIS, si $MIS \neq \emptyset$, sinon retourner "non".

Posons $n = \mathrm{card}(I_e)$ et notons $L_{f,x}$ la courbe de Lorenz discrète (§ 6.2.3.1) associée à un élément x de ^{m} jugé à partir du vrai-critère étendu $g = (f, \geq)$. Rappelons la définition de la dominance de Lorenz généralisée GL :

$$\text{Pour } x, y \in {}^m, \quad y \succcurlyeq_{\mathrm{GL}} x \Leftrightarrow L_{f,y}(k/n) \geq L_{f,x}(k/n),\ \forall\, k \in \{1, \ldots, n\},$$

avec $L_{f,z}(k/n) = \sum_{i=1}^{k} f(z, \sigma_z(i))$, σ_z étant une permutation de n éléments rangeant les valeurs $f(x, k)$ $\forall\, k \in \{1, \ldots, n\}$ en ordre croissant.
Par rapport à GL, la dominance de Lorenz L nécessite en plus que $L_{f,y}(1) = L_{f,x}(1)$.

La dominance de Lorenz généralisée a été utilisée ces dernières années en recherche opérationnelle sur les problèmes de location (voir sur ce point OGRYCZAK [2000]) et en pro-

grammation linéaire multi-objective (voir KOSTREVA & OGRYCZAK [1999]). Ils ramenèrent le problème GL-MOP à un programme multiobjectif agrégé par la dominance de Pareto P-MOP, selon le théorème suivant :

Théorème 3 : Une solution $x \in MIS$ est une solution maximale pour le problème GL-MOP, si et seulement si x est une solution maximale pour le problème P-MOP suivant :

Maximiser $z(k)$ pour tout $k \in \{1, \ldots, n\}$

$$\text{sous les contraintes :} \quad \begin{cases} x \in MIS \\ y(k) = f(x,k) \quad \text{pour tout } k \in \{1,\ldots,n\} \\ z(k) \leq \sum_{i=1}^{k} y(\sigma(i)) \quad \begin{cases} \text{pour tout } k \in \{1,\ldots,n\} \text{ et} \\ \text{pour tout } \sigma \in \mathcal{S}_n \end{cases} \end{cases}$$

avec $\mathcal{S}_n$ l'ensemble des permutations de n éléments.

Cette transformation nous autorise à utiliser toutes les techniques de parcours interactif basées sur les fonctions objectives temporaires dédiées à la dominance de Pareto, pour trouver des solutions maximales pour GL-MOP.

Le troisième groupe de contraintes linéaires comporte $n \cdot n!$ inégalités. Ce nombre d'inégalités peut être considérablement réduit en utilisant les facettes du multipermutoèdre (§ 6.2.3.2).

Rappelons que le multipermutoèdre $P(y)$ est la projection du polytope des matrices bistochastiques d'ordre n sur ^{n} par le vecteur y. Il permet de caractériser la dominance de Lorenz d'après le théorème de Hardy, Littlewood et Polyà (§ 6.2.3.3). Et, comme corollaire de ce théorème, nous avons : $P(y) = \{z \in \ ^{n} \text{ tel que : } z \succcurlyeq_{L} y\}$

La preuve est directe.

Selon les particularités du vecteur y des polyèdres combinatoirement différents[171] sont obtenus. Par exemple, il existe 2 polytopes combinatoirement différents pour le multipermutoèdre : le triangle et l'hexagone. En dimension 3, il en existe 5, l'octaèdre régulier de la Figure 30 en section 6.2.3.2 étant l'un d'eux. Le théorème suivant, dû à BILLERA & SARANGARAJAN [1996], permet de connaître une description des facettes du multipermutoèdre par des inégalités dans le cas général.

Théorème 4 : Le multipermutoèdre $P(y)$ admet la description suivante de ces facettes par des inégalités :

$$P(y) = \left\{ z \in \ ^{n} \text{ tel que : } \sum_{k=1}^{n} z(k) = \sum_{k=1}^{n} y(k) \ \text{ et } \ \sum_{k \in B} z(k) \geq L_y(\mathrm{card}(B)/n) \quad \forall\, B \subset \{1,\ldots,n\} \right\}$$

Pour une description de chaque polytope formant le multipermutoèdre, voir BILLERA & SARANGARAJAN [1996]. De la même manière, il est possible de caractériser la dominance de Lorenz généralisée en un point y, par un polyèdre nommé le **multipermutoèdre généralisé** $GP(y)$. La description de ces facettes par des inégalités se déduit directement de celle du multipermutoèdre :

Corollaire : Une description des facettes du multipermutoèdre généralisé $GP(y)$ par des inégalités correspond à celle du multipermutoèdre $P(y)$, pour laquelle l'opérateur d'égalité de la première contrainte linéaire est remplacé par l'opérateur supérieur ou égal.

[171] Deux polyèdres sont combinatoirement équivalents si l'ensemble des faces des deux polyèdres sont isomorphes. Sinon, ils sont combinatoirement différents. Voir par exemple ZIEGLER [1995] ou BOISSONNAT & YVINEC [1995].

La preuve est directe. En définitive, le troisième groupe d'inégalités du Théorème 3 peut être remplacé par $z \in GP(y)$. Le nombre d'inégalités[172] passe de $n \cdot n!$ à $2^n - 1$. Ce qui est un gain considérable, et nettement plus abordable dans la pratique.

Toujours est-il que le nombre d'inégalités reste exponentiel en la taille des données. Bien que sa résolution puisse être faite par des techniques basées sur la génération de coupes (Branch-and-Cut), pour un gain en performances, d'autres alternatives sont étudiées. D'après le théorème de Hardy, Littlewood & Polyà (§ 6.2.3.3), toute fonction Schur-concave peut être utilisée comme fonction objective temporaire de manière à rechercher des solutions maximales pour la dominance de Lorenz. OGRYCZAK [2000] proposa plusieurs familles de vrai-critères d'agrégation qui identifient des solutions optimales qui sont maximales pour la dominance de Lorenz généralisée. Pour l'instant, elles permettent d'identifier un sous-ensemble de cet ensemble maximal. Cette liste de fonctions objectives agrégatives peut être complétée des familles de vrai-critères présentées en TCS (voir FOSTER & SEN [1997]).

Pour clôturer cette sous-section, donnons un exemple de vrai-critère. Le coefficient de Gini (§ 6.2.2.4) qui, rappelons-le, est une extension de la règle de Lorenz, suggère une description linéaire du problème de pure répartition. Le programme linéaire obtenu considère l'ensemble des solutions réalisables du problème de pure répartition $S(I_{PR})$ auquel il ajoute :

– le critère : Minimiser $\sum_{i \in E} \sum_{\substack{j \in E \text{ et} \\ j > i}} d^+(i,j) + d^-(i,j)$

– les contraintes : $\begin{cases} x(i) - x(j) = d^+(i,j) - d^-(i,j) & \text{pour tout } (i,j) \in Tr(E) \\ d^+(i,j) \geq 0 \quad \text{et} \quad d^-(i,j) \geq 0 & \text{pour tout } (i,j) \in Tr(E). \end{cases}$

avec $Tr(E) = \{(i,j) \in E \times E \text{ tel que : } i < j\}$. Le nombre de variables et contraintes ajoutées reste polynomial (de l'ordre de $\mathrm{card}(E)^2$) en la taille de l'instance du problème de pure répartition.

7.2.2.2. Multipolyèdres et Contraintes Souples

Quelle est la relation entre le polyèdre paramétré caractéristique d'une dominance (appelé **multipolyèdre**) et les contraintes souples associées à cette dominance ? Développons le cas de la dominance de Lorenz :

Associons à chaque individu observé $i \in I_e$ une variable $v(i)$, et notons V l'ensemble de variables obtenues. En considérant pour chaque variable v un domaine $D(v) =$, le multipermutoèdre en un point y est, par définition, l'ensemble des solutions réalisables de la contrainte souple de Lorenz en (surclasse, y) :

$$S(c_{\mathrm{L}}[\text{surclasse}, y], V, D) = P(y) \quad \text{pour tout } y \in \ ^n.$$

Il en est de même de la dominance de Lorenz généralisée : $S(c_{\mathrm{GL}}[\text{surclasse}, y], V, D) = GP(y)$, pour tout $y \in \ ^n$. De la même manière, nous pouvons utiliser le markotope dans le cas d'une dominance de Lorenz multidimensionnelle (§ 6.2.7.2).

Nous pouvons aller plus loin en disant que tout système de contraintes basé sur les préférences $(\mathcal{K}, \{c_{\succcurlyeq}[\alpha, z]\}_{\alpha, z})$, où

(a) $\mathcal{K} = (V, D, C)$ est un système de contraintes décrit en système d'inégalités en nombres mixtes, et

(b) $\{c_{\succcurlyeq}[\alpha, z]\}_{\alpha, z}$ est la contrainte souple finale dont la sémantique en (surclasse, y) forme un polyèdre convexe dans l'espace des critères $\mathcal{D}_V$, décrit par des inégalités linéaires,

peut être transformé en problème multicritère utilisant la dominance de Pareto.

---

[172] La formule de Stirling nous informe que $n!$ $\sqrt{2\pi n}\left(\frac{n}{e}\right)^n$

Cette particularité permet d'élargir l'application des procédures type Branch-and-Bound des fonctions objectives vers les relations binaires de préférence. Schématiquement, chaque fonction objective linéaire est un ensemble de demi-espaces totalement emboîtés, alors que chaque relation binaire de préférence 'linéarisable' est un multipolyèdre, c'est-à-dire un ensemble de polyèdres partiellement emboîtés. Le but de l'optimisation linéaire est d'identifier une solution réalisable se trouvant dans le plus petit (au sens de l'inclusion) demi-espace dont l'intersection avec le domaine réalisable est non vide. Par analogie, le multiobjectif 'linéarisable' cherche à identifier une solution réalisable se trouvant dans un (resp. chaque) plus petit (au sens de l'inclusion) polyèdre dont l'intersection avec le domaine réalisable est non vide.

7.3. Problèmes de Consistance et Propriétés des Contraintes Souples

7.3.1. Décomposition et Filtrage Adaptable pour les PB-Contraintes Souples

Voyons maintenant comment réduire le nombre d'algorithmes de filtrage à concevoir pour les PB-contraintes souples simples et étendues. Pour cela, les propriétés de ces contraintes liées aux attitudes, et présentées au § 5.2.4.1, peuvent être exploitées.

Prenons le cas des PB-contraintes souples simples $\{c_k[\alpha, x]\}_{\alpha, x}$. Pour y fixé, il faudrait établir un algorithme de filtrage pour chaque contrainte associée à une attitude propre. Dans le cadre des relations binaires de préférences, le nombre de d'algorithmes à concevoir se monte à $2^{\mathrm{card}(AF(RBP))} - 2 = 14$. Les dépendances entre les attitudes permettent de réduire ce nombre à 4, c'est-à-dire au nombre d'attitudes fondamentales. Voyons comment :

D'abord rappelons quelques relations existantes entre l'opération de projection et d'autres opérations purement ensemblistes :

Considérons deux sous-ensembles A et B d'un produit cartésien $E = E_1 \times \ldots \times E_n$. Notons proj_i la projection de l'ensemble des parties de E vers l'ensemble des parties de E_i. Alors les relations suivantes sont vérifiées (JAULIN *et al*. [2001], BENHAMOU & OLDER [1997]) :

$$\mathrm{proj}_i(A \cup B) = \mathrm{proj}_i(A) \cup \mathrm{proj}_i(B) \,. \tag{7–36}$$

$$\mathrm{proj}_i(A \cap B) \subseteq \mathrm{proj}_i(A) \cap \mathrm{proj}_i(B) \,. \tag{7–37}$$

Ces relations se généralisent sans difficultés aux sous-ensembles non vides $J \subseteq \{1, \ldots, n\}$:

$$\mathrm{proj}_J(A \cup B) = \mathrm{proj}_J(A) \cup \mathrm{proj}_J(B) \,. \tag{7–38}$$

$$\mathrm{proj}_J(A \cap B) \subseteq \mathrm{proj}_J(A) \cap \mathrm{proj}_J(B) \,. \tag{7–39}$$

Maintenant, donnons quelques précisions sur la notation (voir aussi Annexe A2) : Tout problème de consistance inverse globale GIC reçoit en entrée une contrainte c, un ensemble de variables V et un ensemble de domaines D associé à V, et retourne un ensemble de domaines noté $\mathrm{GIC}(c, V, D)$ associé à V. Par conséquent la notation $\mathrm{GIC}(c, V, D)(v)$ désigne le domaine associé à la variable $v \in V$ de la sortie $\mathrm{GIC}(c, V, D)$. En d'autres termes, $\mathrm{GIC}(c, V, D)(v) = \mathrm{proj}_v(S(c, V, D))$.

Appliquons maintenant tout ceci aux PB-contraintes souples. Nous avons les propositions suivantes :

Proposition 15 : Considérons une PB-contrainte souple simple $\{c_k[\alpha, y]\}_{\alpha, y}$ émise par un individu k sur un ensemble de variables V associé à l'ensemble de domaines D. Pour tout α_1, $\alpha_2 \in \mathcal{PR}(AF)$ et $x \in \mathcal{D}_V$, nous avons :

(a) $\mathrm{GIC}(c_k[\alpha_1 \cup \alpha_2, x], V, D)(v) = \mathrm{GIC}(c_k[\alpha_1, x], V, D)(v) \cup \mathrm{GIC}(c_k[\alpha_2, x], V, D)(v)$, pour tout $v \in V$.

(b) $\mathrm{GIC}(c_k[\alpha_1 \cap \alpha_2, x], V, D)(v) \subseteq \mathrm{GIC}(c_k[\alpha_1, x], V, D)(v) \cap \mathrm{GIC}(c_k[\alpha_2, x], V, D)(v)$, pour tout $v \in V$.

Preuve: Il suffit d'utiliser la relation (a) (resp. la relation (b)) présentée dans les Propriétés 2 en page 152 sur $S(c_k[\alpha_1 \cup \alpha_2, x], V, D)$ (resp. $S(c_k[\alpha_1 \cap \alpha_2, x], V, D)$) suivi de la propriété (7–36) (resp. (7–37)) pour aboutir aux résultats escomptés. ❑

Proposition 16. Considérons une PB-contrainte souple étendue $\{c_k[\alpha, y, i, j]\}_{(\alpha, y, j, j)}$ émise par un individu k sur un ensemble de variables V associé à l'ensemble de domaines D et un ensemble d'individus observés I_e. Pour tout $\alpha_1, \alpha_2 \in \mathcal{PR}(AF)$, $x \in \mathcal{D}_V$ et $i, j \in I_e$, nous avons :

(a) $\mathrm{GIC}(c_k[\alpha_1 \cup \alpha_2, x, i, j], V, D)(v) = \mathrm{GIC}(c_k[\alpha_1, x, i, j], V, D)(v) \cup \mathrm{GIC}(c_k[\alpha_2, x, i, j], V, D)(v)$, pour tout $v \in V$.

(b) $\mathrm{GIC}(c_k[\alpha_1 \cap \alpha_2, x, i, j], V, D)(v) \subseteq \mathrm{GIC}(c_k[\alpha_1, x, i, j], V, D)(v) \cap \mathrm{GIC}(c_k[\alpha_2, x, i, j], V, D)(v)$, pour tout $v \in V$.

Preuve: Il suffit d'appliquer la relation (a) (resp. la relation (b)) présentée dans les Propriétés 3 sur $S(c_k[\alpha_1 \cup \alpha_2, x, i, j], V, D)$ (resp. $S(c_k[\alpha_1 \cap \alpha_2, x, i, j], V, D)$) suivi de la propriété (7–36) (resp. (7–37)) pour aboutir au résultat. ❑

En résumé, les égalités (a) des deux propositions indiquent que l'union conserve la GIC consistance entre les contraintes paramétrées par les attitudes composantes et la contrainte paramétrée par l'attitude composée. Mais l'inverse est faux (inclusions (b) des deux propositions précédentes).

Ainsi, si nous avons un algorithme résolvant le problème GIC pour toute attitude fondamentale alors nous avons aussi un algorithme résolvant le problème GIC pour toutes les autres attitudes. Ce qui monte le nombre d'algorithmes de filtrage à construire à $\mathrm{card}(AF)$. Plus généralement, il suffit de concevoir un algorithme de filtrage établissant un niveau de consistance élevé pour chaque contrainte associée à une attitude fondamentale pour posséder un algorithme avec un fort niveau d'élagage pour toute attitude propre.

Les égalités (a) des deux propositions précédentes donnent des conditions suffisantes pour avoir un algorithme pour chaque contrainte associée à une attitude propre. Mais, la conception d'algorithmes dédiés à chaque contrainte associée à une attitude propre garantit de meilleures performances, voire quelques fois en terme de complexité des problèmes : rien n'empêche que la complexité de deux contraintes associées à des attitudes fondamentales soit élevée alors que celle de la conjonction soit faible.

Une conséquence des inclusions (b) est que, l'exploitation des relations entre les attitudes fondamentales et l'attitude "surclasse" (voir la définition d'une rbp à la sous-section 3.3.1.2), donne seulement une approximation de la consistance GIC si nous effectuons une simple intersection.

Une question importante reste en suspens : Existe-il une manière de réduire la différence entre la partie gauche des formules d'inclusion (b) et sa partie droite ? D'abord, remarquons que la formule (b) de la Proposition 15 consiste à appliquer sur chaque contrainte $c_k[\alpha_q, x]$ avec $q \in \{1, 2\}$, un algorithme GIC à partir du même ensemble de domaines D. Cette opération est équivalente à définir un niveau de consistance très faible, noté CL0 sur le système de contraintes $\mathcal{K}_1 = (V, D, \{c_k[\alpha_1, x], c_k[\alpha_2, x]\})$. Par conséquent, si nous appliquons un niveau de consistance plus élevé (voir VAN HENTENRYCK & DEVILLE [1991], BESSIÈRE *et al.* [1999] pour un complément) comme AC (Annexe A2) sur ce système de contraintes, une meilleure approximation est obtenue. La même remarque est valable pour les PB-contraintes souples étendues. La proposition suivante formalise ces remarques :

Proposition 17. Soit une PB-contrainte souple simple $\{c_k[\alpha, y]\}_{\alpha, y}$ émise par un individu k sur un ensemble de variables V associé à l'ensemble de domaines D. Pour tout $\alpha_1, \alpha_2 \in \mathcal{PR}(AF)$ et $x \in \mathcal{D}_V$, si nous considérons le système de contraintes $\mathcal{K}_1 = (V, D, \{c_k[\alpha_1, x], c_k[\alpha_2, x]\})$ et le niveau de consistance CL plus élevé que CL0, alors :

$$\mathrm{GIC}(c_k[\alpha_1 \cap \alpha_2, x], V, D)(v) \subseteq D_{\mathrm{CL}(1)}(v) \subseteq D_{\mathrm{CL}0}(v) \quad \text{pour tout } v \in V,$$
avec $D_{\mathrm{CL}(1)}$ l'ensemble des domaines du système de contraintes $\mathcal{R}_{\mathrm{CL}(1)} = \mathrm{CL}(\mathcal{R}_1)$ et
$$D_{\mathrm{CL}0}(v) = \mathrm{GIC}(c_k[\alpha_1, x], V, D)(v) \cap \mathrm{GIC}(c_k[\alpha_2, x], V, D)(v)$$

L'égalité est obtenue lorsque CL = GIC.

Preuve: Cette preuve est directe parce que $S(c_{\succcurlyeq}[\alpha, x], D) = S(\mathcal{R}_1)$. En effet, c'est la définition des systèmes de contraintes qui est directement exploitée ici : Une solution réalisable d'un système de contraintes est une solution réalisable pour chacune de ses contraintes. Ce qui est équivalent à dire que l'ensemble des solutions réalisables d'un système de contraintes est égal à l'intersection des ensembles des solutions réalisables de ses contraintes. Par conséquent, l'inclusion inverse est obtenue si le niveau de consistance CL sur $\mathcal{R}_1$ est égal à GIC. ❑

Proposition 18. Soit une PB-contrainte souple étendue $\{c_k[\alpha, y, i, j]\}_{(\alpha, y, j, j)}$ émise par un individu k sur un ensemble de variables V associé à l'ensemble de domaines D et un ensemble d'individus observés I_e. Pour tout $\alpha_1, \alpha_2 \in \mathcal{PR}(AF)$, $x \in \mathcal{D}_V$ et $i, j \in I_e$, si nous considérons le système de contraintes $\mathcal{R}_1 = (V, D, \{c_k[\alpha_1, x, i, j], c_k[\alpha_2, x, i, j]\})$ et le niveau de consistance CL plus élevé que CL0, alors :

$$\mathrm{GIC}(c_k[\alpha_1 \cap \alpha_2, x, i, j], V, D)(v) \subseteq D_{\mathrm{CL}(1)}(v) \subseteq D_{\mathrm{CL}0}(v) \quad \text{pour tout } v \in V,$$
avec $D_{\mathrm{CL}(1)}$ l'ensemble des domaines du système de contraintes $\mathcal{R}_{\mathrm{CL}(1)} = \mathrm{CL}(\mathcal{R}_1)$ et
$$D_{\mathrm{CL}0}(v) = \mathrm{GIC}(c_k[\alpha_1, x, i, j], V, D)(v) \cap \mathrm{GIC}(c_k[\alpha_2, x, i, j], V, D)(v)$$

L'égalité est obtenue lorsque CL = GIC.

Preuve: La preuve est la même que pour la Proposition 17. ❑

Les deux propositions précédentes seront utilisées pour définir des algorithmes de filtrage pour différentes règles d'agrégation. Selon le niveau de consistance CL utilisé, nous obtiendrons un algorithme plus ou moins performant. Ces algorithmes de filtrage paramétrés par un (ou plusieurs) niveau(x) de consistance CL seront appelés **algorithmes de filtrage adaptables**. Remarquons que cette notion d'adaptabilité des algorithmes de filtrage rejoint beaucoup celui de " *customizable framework for inverse local consistency*" introduit par VERFAILLIE *et al*. [1999].

7.3.2. Filtrage pour les VC-Contraintes Souples

Comment concevoir un algorithme de filtrage pour un vrai critère ? La méthode la plus répandue en programmation par contraintes utilise l'arithmétique sur les intervalles[173]. Dans ce cadre, la contrainte de somme sera présentée au § 7.4.3.4.

Mais il est généralement plus efficace de combiner ces contraintes souples avec d'autres contraintes dures (RÉGIN & RUEHER [2000], RÉGIN *et al*. [2001]) ou souples de l'application. Dans un tel cadre, des algorithmes de filtrage peuvent être basés sur les algorithmes de RO, comme la programmation linéaire (voir Annexe A2) pour l'évaluation de sous-problèmes linéaires.

[173] Voir par exemple, BENHAMOU & OLDER [1997], HICKEY *et al*. [2001], JAULIN *et al*. [2001]. Pour un état de l'art récent, voir le rapport de synthèse du projet COCONUT à l'adresse : http://www.mat.univie.ac.at/~neum/glopt/coconut/index.html.

7.4. Consistance pour les Contraintes Globales Souples de Quelques Règles d'Agrégation

Nous présentons maintenant un certain nombre d'algorithmes de filtrage pour règles d'agrégation, dont certains ont la particularité d'être adaptables au problème étudié.

Les règles d'agrégation étudiées sont la dominance de Pareto sur un ensemble de vrai-critères (§ 7.4.1), la règle de Pareto (§ 7.4.2), la dominance de Lorenz (§ 7.4.3), la dominance de Lorenz généralisée (§ 7.4.3), la règle de Pareto sur les positions (§ 7.4.4.1), la règle de Pareto conditionnée par l'Identité (§ 7.4.4.2) et enfin, la règle d'équité de Suppes-Sen (§ 7.4.5).

Comme nous l'avons vu précédemment, toutes ces règles d'agrégation ont la particularité de satisfaire un des axiomes d'indépendance (§ 4.2.2.2) : IIPC ou IIIC. De plus, nous confectionnons des algorithmes de filtrage indépendant du paramètre solution des contraintes constituant les PB-contraintes souples dérivées de ces règles d'agrégation. En utilisant alors les résultats (conséquences de la Proposition 15(a) et la Proposition 16(a)) de la sous-section 7.3.1, il suffit de construire 4 algorithmes de filtrage pour décrire complètement le problème de consistance d'une règle d'agrégation. Nous construisons donc en priorité des algorithmes de filtrage pour les attitudes fondamentales.

7.4.1. Global Inverse Consistance pour la Dominance de Pareto sur un Profil de Vrai-Critères

La dominance Pareto est une agrégation de fonctions objectives individuelles. Ses préférences individuelles sont représentées par des VC-contraintes souples individuelles, et ses préférences collectives sont modélisées par une PB-contrainte souple appelée contrainte souple de dominance de Pareto.

Considérons un ordre total $(E, \geq)$, et un profil de vrai-critères $\mathcal{F} = \{f_1, \ldots, f_n\}$ émis par les individus $I_e = \{1, \ldots, n\}$, chaque fonction objective f_i étant à valeur dans E. A chaque fonction objective f_i est associée une VC-contrainte souple c_i et sa variable valeur v_i. L'ensemble des variables valeurs est noté V, et toute variable v de V prend ses valeurs dans E ($\Leftrightarrow D(v) \subseteq E$).

La contrainte souple de dominance de Pareto $c_P[\alpha, x]((E, \geq), \mathcal{F})$ en l'attitude α et la solution x associée au couple $((E, \geq), \mathcal{F})$ est notée plus simplement $c_P[\alpha, x]$.

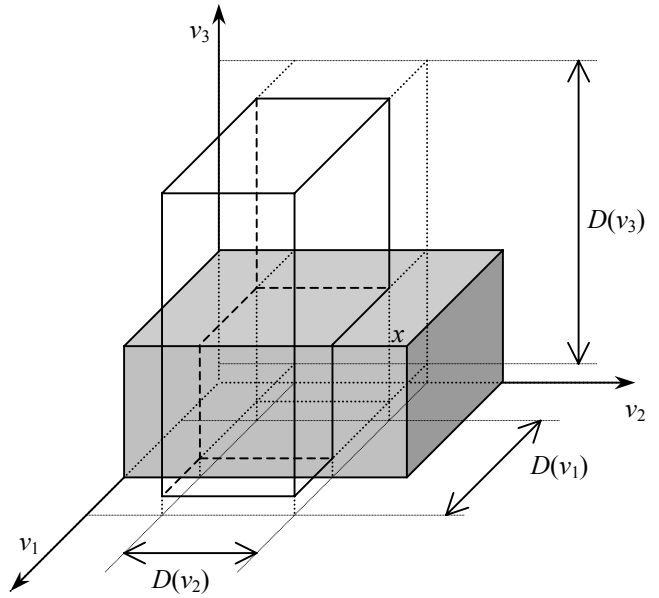
La sémantique de $c_P[\alpha, x]$ peut être définie entièrement à partir des variables valeurs V, la modification de celles-ci modifiant l'ensemble des solutions réalisables des VC-contraintes souples individuelles. Par conséquent, par la suite, nous ferons référence au profil $\mathcal{F}$ des préférences individuelles uniquement par les variables valeurs V et les domaines associés D.

Remarquons que la contrainte souple de dominance de Pareto peut être généralisée en considérant un ensemble totalement préordonné $(E(v), \geq_v)$ par variable $v \in V$ (par exemple un ensemble discret $E(v)$ pour v, un ensemble continu pour une autre variable, etc.) sans modifier notablement notre discours. Il suffit de remplacer E (respectivement, $\geq$) par $E(v)$ (respectivement, $\geq_v$), le contexte d'utilisation indiquant clairement la variable cible v. De plus, l'utilisation de E ou de tout ensemble de classes d'équivalence sur E pour la relation d'équivalence associée à $\geq$, n'a aucune influence notable sur la sémantique de la contrainte.

Étudions maintenant quelques algorithmes de filtrage pour cette contrainte souple. En tenant compte de la Proposition 15(a), il suffit de concevoir un algorithme établissant la GIC pour chaque contrainte associé à une attitude fondamentale α. Avant de s'exécuter au § 7.4.1.2, nous allons d'abord présenter puis généraliser le travail de GAVANELLI sur l'attitude {mieux, incomparable} au § 7.4.1.1.

7.4.1.1. Algorithme de Filtrage Global de GAVANELLI pour l'Attitude {mieux, incomparable}

La contrainte souple de dominance de Pareto en l'attitude α = {mieux, incomparable} et en une solution x quelconque a été étudiée par GAVANELLI [2002*a*] et [2002*b*] dans le cas des intervalles bornés. La Figure 33 nous donne un exemple de cette contrainte dans le cas de trois vrai-critères, E étant égal à .



Légende :
- les trois vrai-critères sont représentés par les variables $\{v_1, v_2, v_3\}$ = V dont les domaines sont des intervalles de .
- Le produit cartésien des domaines $\mathcal{D}_V$ est le rectangle debout transparent, dont les cotes correspondent aux domaines.
- Le rectangle gris désigne les solutions qui sont rejetées par la contrainte mole de Pareto.
- L'intersection entre le rectangle gris et le rectangle transparent désigne les solutions de $\mathcal{D}_V$ inconsistantes avec la contrainte mole de Pareto.

Figure 47 : Découpe de l'espace des critères ^{3} engendré par la contrainte souple de dominance de Pareto en ({mieux, incomparable}, x).

GAVANELLI [2002*a* Property 5, p. 123] et [2002*b*] donne une condition nécessaire et suffisante pour qu'une valeur de variable soit inconsistante avec la contrainte de dominance de Pareto en ({mieux, incomparable}, x). Ce résultat est présenté par le théorème suivant :

Théorème 5 : Notons y_i = (max($D(v_1)$), …, max($D(v_{i-1})$), min($D(v_i)$), max($D(v_{i+1})$), …, max($D(v_n)$)) un élément de E^n, c'est-à-dire la meilleure solution pour tous les individus sauf pour i.
Alors D peut être réduit s'il existe un i tel que $y_i < x$ (⇔ $y_i(v) \leq x(v)$, pour tout $v \in V$ et il existe une variable v telle que : $y_i(v) < x(v)$).

Preuve: La valeur $d \in D(v_i)$, avec $1 \leq i \leq n$, est inconsistante avec la contrainte de Pareto en ({mieux, incomparable}, x) ⇔ les solutions de l'ensemble $\mathcal{D}_{(v,\, d)} = D(v_1) \times \ldots \times D(v_{i-1}) \times ([\min(D(v_i)), d] \cap D(v_i)) \times D(v_{i+1}) \times \ldots \times D(v_n)$ sont inconsistantes avec cette contrainte ; En particulier, la solution (max($D(v_1)$), …, max($D(v_{i-1})$), d, max($D(v_{i+1})$), …, max($D(v_n)$)) Pareto-domine toutes les solutions de $D_{(v,\, d)}$, et notamment y_i. ❑

De ce théorème, GAVANELLI [2002] déduisit un algorithme en O(n), en testant[174] si pour toute variable $v \in V$, max($D(v)$) est inférieur ou égal à la valeur respective $x(v)$. Dans ce cas, il n'y a aucune solution réalisable (⇔ $y \preccurlyeq_P x$ pour tout $y \in \mathcal{D}_V$). Sinon si max($D(v)$) est inférieur ou égal à la valeur respective $x(v)$ sauf pour exactement une variable v_i, pour laquelle le minimum de son domaine $D(v_i)$ est inférieur (ou égal si min($D(v_i)$) $\in D(v_i)$) à la valeur associée $x(v_i)$, alors toute valeur de $D(v_i)$ bornée supérieurement par $x(v_i)$ peut être éliminée. Cet algorithme est précisé ci-dessous.

[174] GAVANELLI utilise un algorithme plus complexe, afin d'optimiser la gestion d'un ensemble de contraintes moles de Pareto en {mieux, incomparable} utilisé dans son algorithme d'énumération présenté au § 7.1.1.

```
GICPareto{mieux, incomparable}((E, ≥) : ensemble totalement ordonné,
                                V : ensemble de variables, D : ensemble de domaines,
                                x : solution) : retourner {ensemble de domaines, non}
début
   v_b ∈ V ∪ {ε_V} ← ε_V        désigne la variable v dont le domaine a une borne supérieure plus grande que
                                x(v), si elle existe et est unique, sinon vaut ε_V. Elle est initialisée à ε_V.
   pour tout ( v ∈ V ) faire
      si ( max(D(v)) > x(v) ) alors[175]
         si (v_b = ε_V ) alors     v_b ← v
         sinon     retourner D     Pas de filtrage : toutes les valeurs de domaines conduisent à
                                   une solution réalisable pour la contrainte.
         fin si
      fin si
   fin pour
   si ( v_b = ε_V ) alors     retourner non     fin si     Il n'existe pas de solution réalisable.
            Ici, v_b ≠ ε_V et désigne l'unique variable dont le domaine comporte des éléments supérieurs à x(v_b).
   D(v_b) ← D(v_b) \ [min(D(v_b)), x(v_b)]
   retourner D     La propagation s'est bien passée.
fin GICPareto{mieux, incomparable}
```

Algorithme 6 : Filtrage de la contrainte souple de dominance de Pareto en [{mieux, incomparable}, *x*], établissant un niveau de consistance GIC.

7.4.1.2. Filtrage Global pour les Contraintes Associées aux Attitudes Fondamentales

La contrainte de dominance de Pareto en l'attitude {mieux, incomparable} n'est pas suffisante pour décrire toute la sémantique de cette contrainte souple en chaque attitude. Rappelons que cette description est nécessaire pour concevoir des algorithmes de recherche de solutions plus performants (§ 7.1.2 et § 7.1.3), et concevoir des modèles de préférence plus élaborés (voir les hiérarchies agrégatives au § 4.2.1.5). La Proposition 15(a) et ses conséquences informent qu'il est suffisant de concevoir un algorithme établissant la GIC pour la contrainte de dominance de Pareto en chaque attitude fondamentale, pour décrire entièrement un algorithme établissant la GIC pour chaque attitude propre.

La contrainte de dominance de Pareto en (indifférent, x) est la plus simple. Un algorithme de filtrage pour la GIC consiste à instancier les domaines D aux valeurs de la solution x, puis de les retourner, si et seulement si $x \in \mathcal{D}_V$.

❖ La Préférence Stricte et l'Aversion Stricte

L'algorithme suivant permet d'établir la consistance inverse globale (GIC) pour la contrainte de dominance de Pareto en (mieux, x).

```
GICPareto{mieux}((E, ≥) : ensemble totalement ordonné, V : ensemble de variables,
                 D : ensemble de domaines, x : solution):
                                                retourner {ensemble de domaines, non}
début
   v_b ∈ V ∪ {ε_V} ← ε_V        désigne la variable v dont le domaine a une borne supérieure plus grande que
                                x(v), si elle existe, sinon vaut ε_V. Elle est initialisée à ε_V.
```

[175] Si l'ensemble est discret, $\max(D(v)) \in D(v)$; sinon, dans le cas continu, il existe toujours un élément de $D(v)$ strictement compris entre $\max(D(v))$ et $x(v)$.

unique : Booléen ← non vaut "oui" s'il existe une variable v unique dont le domaine a une borne supérieure plus grande que $x(v)$; sinon vaut "non". Elle est initialisée à "non".

pour tout ($v \in V$) **faire**

$D(v) \leftarrow D(v) \cap [x(v), \max(D(v))]$

si ($D(v) = \varnothing$) **alors** **retourner** non **fin si** Il n'existe pas de solution réalisable.

si ($D(v) \neq \{x(v)\}$) **alors**

si ($v_b = \varepsilon_V$) **alors** unique ← oui ; $v_b \leftarrow v$

sinon unique ← non

fin si

fin si

fin pour

si ($v_b \neq \varepsilon_V$ **et** unique = oui) **alors** $D(v_b) \leftarrow D(v_b) \setminus \{x(v_b)\}$ **fin si**

Il existe une unique variable v_b dont le domaine est différent de $x(v_b)$.

si ($v_b = \varepsilon_V$ **et** unique = non) **alors** **retourner** non **fin si** Ici, $\mathcal{D}_V = \{x\}$ ou $V = \varnothing$.

retourner D La propagation s'est bien passée.

fin GICPareto{mieux}

Algorithme 7 : Filtrage de la contrainte de dominance de Pareto en [mieux, x], établissant l'inverse consistance globale (GIC).

Étant données une variable v_i de V et une valeur d de $D(v_i)$, notons :

$y_{i,d} = (\max(D(v_1)), \ldots, \max(D(v_{i-1})), d, \max(D(v_{i+1})), \ldots, \max(D(v_n)))$, et

$\mathcal{D}_{(v,d)} = D(v_1) \times \ldots \times D(v_{i-1}) \times ([\min(D(v_i)), d] \cap D(v_i)) \times D(v_{i+1}) \times \ldots \times D(v_n)$.

L'algorithme GICPareto{mieux} établit la consistance inverse globale, car elle supprime les valeurs d du domaine des variables v_i telle que : soit $y_{i,d}(v) < x(v)$ pour au moins une variable v de V, soit $y_{i,d}(v) = x(v)$ pour tout v de V. Ce qui signifie que non$[y_{i,d} \succ_P x]$. Comme $y_{i,d} \succcurlyeq_P y$ pour tout $y \in \mathcal{D}_{(v,d)}$, donc par transitivité (voir la table de transitivité des préordres partiels à l'Annexe B1) nous avons non$[y \succ_P x]$.

Le cas de l'attitude α = pire est obtenu directement à partir du cas où α = mieux. Un algorithme établissant la consistance inverse globale pour la contrainte de dominance de Pareto en [pire, x] s'obtient en utilisant l'Algorithme 6 et en remplaçant l'instruction « $D(v) \leftarrow D(v) \cap [x(v), \max(D(v))]$ » par « $D(v) \leftarrow D(v) \cap [\min(D(v)), x(v)]$ ».

❖ L'Incomparabilité

L'algorithme suivant permet d'établir la consistance inverse globale (GIC) pour la contrainte de dominance de Pareto en (incomparable, x).

GICPareto{incomparable} ($(E, \geq)$: ensemble totalement ordonné, V : ensemble de variables, D : ensemble de domaines, x : solution) :

retourner {ensemble de domaines, non}

début

si (card(V) < 2) **ou** ($\mathcal{D}_V = \{x\}$) **alors** **retourner** non **fin si**

si (card(V) = 2) **alors** $D(v_1) \leftarrow D(v_1) \setminus \{x(v_1)\}$; $D(v_2) \leftarrow D(v_2) \setminus \{x(v_2)\}$ **fin si**

en notant $V = \{v_1, v_2\}$

si ($\max(D(v)) \leq x(v)$, **pour tout** ($v \in V$)) **alors** **retourner** non **fin si**

si ($\min(D(v)) \geq x(v)$, **pour tout** ($v \in V$)) **alors** **retourner** non **fin si**

si (**il existe un unique** ($v \in V$) **tel que** $\min(D(v)) < x(v) < \max(D(v))$) **alors**

si ($\max(D(v_b)) \leq x(v_b)$, **pour tout** ($v_b \in V \setminus \{v\}$)) **alors**

$D(v) \leftarrow D(v) \setminus [\min(D(v)), x(v)]$

sinon si ($\min(D(v_b)) \geq x(v_b)$, **pour tout** ($v_b \in V \setminus \{v\}$)) **alors**

$D(v) \leftarrow D(v) \setminus [x(v), \max(D(v))]$

```
    fin si
  fin si
  retourner D
fin GICPareto{incomparable}
```

Algorithme 8 : Filtrage de la contrainte de Pareto[incomparable, x], établissant l'inverse consistance globale (GIC) pour cette contrainte.

Remarques 3. *(Considérations de Codage)* Les algorithmes précédents sont assez généraux pour ne considérer aucune hypothèse sur le codage des domaines. Par conséquent, pour connaître exactement sa complexité, le codage utilisé pour chaque domaine doit être connu. Concernant les trois codages considérés précédemment (§ 2.1.4.2) : l'ensemble explicite, l'intervalle et l'union finie d'intervalles sur un ensemble discret totalement ordonné, les opérations sur les ensembles utilisées (union, intersection, max, min, complémentaire) restent polynomiales en la taille des domaines. Ce qui conduit à des algorithmes de filtrage polynomiaux en la taille de l'instance. Ainsi, si les domaines sont codés par les intervalles, les algorithmes précédents ont une complexité d'au plus $O(n)$ avec $n = \text{card}(V)$. Si les domaines sont des ensembles explicites, les algorithmes précédents ont une complexité d'au plus $O(n + q)$, avec q = le nombre d'éléments du plus grand domaine de D.

Par contre, si un algorithme de filtrage utilise des opérations arithmétiques sur les ensembles, l'union fini d'intervalles ne permet pas de conserver la polynomialité comme nous le verrons au § 7.4.3.4.

(Systèmes de Contraintes Continus) Ces algorithmes de filtrage sont aussi valables pour les domaines continus. De plus, si les domaines continus sont codés par des intervalles ou des unions finis d'intervalles, alors la précédente remarque sur le codage reste aussi valable.

Préoccupons nous maintenant de la conception d'algorithmes de filtrage pour la contrainte souple associée aux préférences collectives de la règle de Pareto, cette contrainte étant plus générale que la contrainte de dominance de Pareto.

7.4.2. Inverse Consistance Adaptable pour la Règle de Pareto

La règle de Pareto est une règle d'agrégation de relations binaires de préférences individuelles en une relation binaire de préférence collective. Les préférences individuelles et collectives sont modélisées par des PB-contraintes souples simples, notées respectivement $\{c_i[\alpha, x]\}_{\alpha, x}$ pour tout $i \in I_e$, et $\{c_P[\alpha, x]\}_{\alpha, x}$. La contrainte souple collective est appelée contrainte souple de Pareto, et chacun de ces éléments est appelé contrainte de Pareto.

En suivant les propriétés des PB-contraintes souples simples (§ 7.3.1) nous présentons ici un algorithme adaptable pour chaque contrainte de Pareto en les différentes attitudes fondamentales.

Pour simplifier l'écriture, nous supposons que l'ensemble de variables V est commun à la fois aux contraintes souples individuelles $\{c_i[\alpha, x]\}_{\alpha, x}$ et à la contrainte souple collective $\{c_P[\alpha, x]\}_{\alpha, x}$. De plus, D désigne l'ensemble des domaines initiaux.

7.4.2.1. Filtrage Global et Adaptable pour l'Indifférence

L'ensemble des solutions réalisables de la contrainte de Pareto en (indifférent, x) est défini comme suit :

$$\begin{aligned} S(c_P[\text{indifférent}, x], V, D) &= \{y \in \mathcal{D}_V \text{ tel que : } y \simeq_P x\} \\ &= \{y \in \mathcal{D}_V \text{ tel que : } y \simeq_i x \text{ pour tout } i \in I_e\} \\ &= \{y \in \mathcal{D}_V \text{ tel que : } y \in S(c_i[\text{indifférent}, x], V, D) \text{ pour tout } i \in I_e\} \end{aligned}$$

En définitive, $S(c_{\mathrm{P}}[\text{indifférent}, x], V, D) = \bigcap_{i \in I_e} S(c_i[\text{indifférent}, x], V, D)$. (7–40)

L'égalité (7–40) permet d'exprimer la sémantique de la contrainte de Pareto en (indifférent, x) en fonction de la sémantique des contraintes souples individuelles en (indifférent, x). L'application de la Proposition 15(b) permet de concevoir un algorithme établissant un niveau de consistance inférieur au GIC. Nous pouvons aller au-delà dans la réflexion, en considérant le système de contraintes suivant :

$$\mathcal{R}_{\mathrm{P}(\text{indifférent}, x)} = (V, D, \{c_i[\text{indifférent}, x] \text{ tel que : } i \in I_e\})$$

C'est le système de contraintes formé uniquement des contraintes souples individuelles en (indifférent, x). Nous avons alors la proposition suivante :

Proposition 19 . $S(c_{\mathrm{P}}[\text{indifférent}, x], V, D) = S_{\mathrm{CSP}}(\mathcal{R}_{\mathrm{P}(\text{indifférent}, x)})$

Preuve: Elle est donnée par la troisième égalité des extensions de $S(c_{\mathrm{P}}[\text{indifférent}, x], V, D)$ donné précédemment. ❑

Ainsi, tout algorithme résolvant un problème de consistance sur $\mathcal{R}_{\mathrm{P}(\text{indifférent}, x)}$ est un algorithme de filtrage pour $c_{\mathrm{P}}[\text{indifférent}, x]$. Chaque problème de consistance sur $\mathcal{R}_{\mathrm{P}(\text{indifférent}, x)}$ est équivalent à un problème de consistance sur $c_{\mathrm{P}}[\text{indifférent}, x]$. Finalement, $\mathrm{GIC}(c_{\mathrm{P}}[\text{indifférent}, x], D) = \mathrm{GIC}(\mathcal{R}_{\mathrm{P}(\text{indifférent}, x)})$. (voir aussi la Proposition 17)

Le développement qui vient d'être fait pour l'indifférence est valable aussi pour le surclassement et le sousclassement, en remplaçant 'indifférent' par respectivement 'surclasse' et 'sousclasse'.

7.4.2.2. Filtrage Global et Adaptable pour la Préférence Stricte et l'Aversion Stricte

L'ensemble des solutions réalisables de la contrainte de Pareto en (mieux, x) est défini comme suit :

$$\begin{aligned}
S(c_{\mathrm{P}}[\text{mieux}, x], V, D) &= \{y \in \mathcal{D}_V \text{ tel que : } y \succ_{\mathrm{P}} x\} \\
&= \{y \in \mathcal{D}_V \text{ tel que : } y \succcurlyeq_i x \text{ pour tout } i \in I_e \text{ et il existe } i \in I_e \text{ avec } y \succ_i x\} \\
&= \left\{ y \in \mathcal{D}_V \text{ tel que : } \left| \begin{array}{l} y \in S(c_i[\text{surclasse}, x], V, D) \quad \forall i \in I_e, \text{et} \\ \text{il existe } i \in I_e \text{ avec } y \in S(c_i[\text{mieux}, x], V, D) \end{array} \right. \right\} \\
&= \bigcup_{i \in I_e} \left(S(c_i[\text{mieux}, x], V, D) \cap \left(\bigcap_{j \in I_e \setminus \{i\}} S(c_i[\text{surclasse}, x], V, D) \right) \right).
\end{aligned}$$

Cette égalité permet d'exprimer la sémantique de la contrainte de Pareto en (mieux, x) en fonction des contraintes souples individuelles en (mieux, x) et (surclasse, x). L'application des résultats du § 7.3.1 sur les égalités précédentes, permet de prouver la proposition suivante :

Proposition 20 . $S(c_{\mathrm{P}}[\text{mieux}, x], V, D) = \bigcup_{i \in I_e} \left(S_{\mathrm{CSP}}(\mathcal{R}_{\mathrm{P}(\text{mieux}, x, i)}) \right)$, avec :

$$\mathcal{R}_{\mathrm{P}(\text{mieux}, x, i)} = (V, D, \{c_i[\text{mieux}, x]\} \cup \{c_j[\text{surclasse}, x] \text{ tel que : } j \in I_e \setminus \{i\}\}.$$

Cette proposition définit la sémantique de la contrainte de Pareto en (mieux, x) comme étant une union des ensembles de solutions réalisables de $\mathrm{card}(I_e)$ CSP sur des systèmes de contraintes formés de $\mathrm{card}(I_e)$ contraintes. Un algorithme adaptable peut être déduit en établissant un niveau de consistance sur chaque système de contraintes $\mathcal{R}_{\mathrm{P}(\text{mieux}, x, i)}$, puis en effectuant l'union des domaines d'une même variable. Ce qui s'écrit en pseudo-code :

`FilterPareto{`mieux`}`(I_e: ensemble des individus concernés, V: ensemble de variables,
D: ensemble de domaines, $\{\{c_i[\alpha, y]\}_{\alpha, y}, \forall\ i \in I_e\}$: ensemble de PB-contraintes souples,
x: solution) : **retourner** {ensemble de domaines, non}
début
$Df \in \{D_0$ tel que : $D_0(v) \subseteq D(v)\}$ ensemble de domaines indexé par V; il reçoit itérativement l'union des domaines obtenus par filtrage.
$\mathcal{K}_{\texttt{Filtre}} \in$ {système de contraintes} $\cup$ {non}
pour tout ($v \in V$) **faire** $Df(v) \leftarrow \varnothing$ **fin pour tout** initialisation
pour tout ($i \in I_e$) **faire**
$\mathcal{K}_{\texttt{Filtre}} \leftarrow \texttt{Filtrer}(\mathcal{K}_{\text{P(mieux}, x, i)})$ appliquer un algorithme de filtrage adaptable sur le système de contraintes identifié par i.
si ($\mathcal{K}_{\texttt{Filtre}} \neq$ non) **alors** notons $\mathcal{K}_{\texttt{Filtre}} = (V, D_{\texttt{Filtre}}, C_{\texttt{Filtre}})$
pour tout ($v \in V$) **faire** $Df(v) \leftarrow Df(v) \cup D_{\texttt{Filtre}}(v)$ **fin pour tout**
fin si
fin pour tout
si ($Df(v) = \varnothing, \forall\ v \in V$) **alors** **retourner** non **sinon** **retourner** Df **fin si**
fin `FilterPareto{`mieux`}`

Algorithme 9. **Algorithme de filtrage adaptable pour la Contrainte de Pareto en (mieux, *x*).**

Notons n = card(I_e). En termes de complexité algorithmique, cet algorithme nécessite de résoudre n problèmes de consistance, à raison d'un par système de contraintes formé de n contraintes.

Les algorithmes adaptables ont la particularité de définir des systèmes de contraintes où un niveau de consistance est recherché via des algorithmes de filtrage généraux `Filtrer`.
`Filtrer` : établit un niveau de consistance à préciser sur un système de contraintes. `Filtrer` correspond à n'importe quel algorithme de consistance. Par conséquent, un algorithme très général établissant l'arc-consistance ou tout autre niveau d'inverse consistance peut être utilisé. Des algorithmes plus spécifiques au problème peuvent aussi être exploités. Si un algorithme établissant la GIC est utilisé, alors `FilterPareto{`mieux`}` est un algorithme établissant une GIC sur le système formé de la seule contrainte de Pareto.

Une proposition et un algorithme similaires existent pour la contrainte de Pareto en (pire, x). Pour cela, il suffit de remplacer "mieux" par "pire", et "surclasse" par "sousclasse".

7.4.2.3. Filtrage Global et Adaptable pour l'Incomparabilité

De la même manière, la relation suivante est déduite entre l'ensemble des solutions réalisables de la contrainte de Pareto en (incomparable, x) et les contraintes souples individuelles :

$$S(c_{\text{P}}[\text{incomparable}, x], V, D) = \left(\bigcup_{i \in I_e} S(c_i[\text{incomparable}, x], V, D)\right) \cup \left(\bigcup_{i \in I_e} \bigcup_{j \in I_e \setminus \{i\}} S(c_i[\text{mieux}, x], V, D) \cap S(c_j[\text{pire}, x], V, D)\right).$$

Cette égalité exprime que la sémantique de la contrainte de Pareto en (incomparable, x) peut être formulée en fonction de la sémantique des contraintes souples individuelles en (incomparable, x), (mieux, x) et (pire, x).

Comme précédemment, une proposition et des algorithmes adaptables peuvent être déduits de cette égalité. Nous ne le faisons pas ici. Notons néanmoins que tout algorithme adaptable basé sur cette égalité nécessite la résolution de $n \cdot (n - 1) / 2$ problèmes de consistance à raison

d'un par système de contraintes formé de deux contraintes, et de résoudre n problèmes de consistance sur des contraintes.

7.4.3. Consistance et Dominance de Lorenz

Maintenant, qu'en est-il des préférences de type sympathie étendue ? Les contraintes souples associées à la dominance de Lorenz en sont de bons exemples.

7.4.3.1. Contraintes Souples de Lorenz et de Lorenz Généralisée

Pour illustrer la conception d'algorithmes de consistance pour une règle d'agrégation étendue, nous présentons ci-dessous le cas de la dominance de Lorenz L et de la dominance de Lorenz généralisée GL.

De façon générale, par analogie aux dominances qu'elles représentent, la **contrainte souple de Lorenz** $\{c_{\mathrm{L}}[\alpha, z]\}_{\alpha, z}$ et la **contrainte souple de Lorenz généralisée** $\{c_{\mathrm{GL}}[\alpha, z]\}_{\alpha, z}$ sont définies à partir d'une VC-contrainte souple étendue $\{c_f[i]\}_i$ modélisant un vrai-critère étendu f. Pour simplifier la notation, nous ne mentionnons la VC-contrainte souple étendue que via son ensemble de variables valeurs noté V associant une variable valeur à chaque individu observé (voir § 5.2.3).

Étant issues du même vrai-critère, les variables v de V ont un domaine $D(v)$ inclus dans un même ensemble totalement ordonné : $D(v) \subseteq B$ pour tout $v \in V$. De plus, la dominance de Lorenz et son homologue généralisée nécessitent que la structure mathématique minimale de B soit plus riche pour permettre les additions par exemple. Ainsi, l'ensemble $(B, +, \geq_B)$ possède une structure minimale de quasi-groupe additif canoniquement ordonné[176] comme ($^{+}$, +, ≥) et (, +, ≥). Mais, nous nous placerons généralement directement dans ($^{+}$, +, ≥) et (, +, ≥) en précisant lorsque nécessaire les particularités d'un ensemble discret ou continu.
Posons n = card(V), et notons L_z la courbe de Lorenz discrète (§ 6.2.3.1)[177] associée à un élément z de $\mathcal{D}_V$. Les sémantiques des contraintes souples sont alors, pour tout $x \in \mathcal{D}_V$:

– pour $\alpha \in$ {indifférent, surclasse, sousclasse} :

$$S(c_{\mathrm{GL}}[\alpha, x], D, V) = \{y \in \mathcal{D}_V \text{ tel que : } y\ \alpha_{\mathrm{GL}}\ x\}$$
$$= \{y \in \mathcal{D}_V \text{ tel que : } L_y(i)\ \alpha\ L_x(i),\ \forall\ i \in \{1, \ldots, n\}\ \}$$
$$S(c_{\mathrm{L}}[\alpha, x], D, V) = \{y \in \mathcal{D}_V \text{ tel que : } y\ \alpha_{\mathrm{L}}\ x\}$$
$$= \{y \in \mathcal{D}_V \text{ tel que : } L_y(n) = L_x(n) \quad \text{et} \quad y\ \alpha_{\mathrm{GL}}\ x\}$$

– pour α = mieux :

$$S(c_{\mathrm{GL}}[\text{mieux}, x], V, D) = \left\{ y \in \mathcal{D}_V \text{ tel que} : \left| \begin{array}{l} \text{pour tout } i \in \{1, \ldots, n\},\ \ L_x(i) \geq L_y(i) \\ \text{il existe } 1 \leq j \leq n,\ \text{avec}:\ L_x(j) > L_y(j) \end{array} \right. \right\}$$

$$S(c_{\mathrm{L}}[\text{mieux}, x], V, D) = \{y \in \mathcal{D}_V \text{ tel que : } L_y(n) = L_x(n) \quad \text{et} \quad y \in S(c_{\mathrm{GL}}[\text{mieux}, x], V, D)\}$$

– La sémantique pour α = pire est obtenue à partir de celle de mieux en changeant le sens des inégalités ;

– pour α = incomparable :

$$S(c_{\mathrm{L}}[\text{incomparable}, x], V, D) = \mathcal{D}_V \setminus (\ S(c_{\mathrm{L}}[\text{surclasse}, x], V, D) \cup S(c_{\mathrm{L}}[\text{sousclasse}, x], V, D)\ )$$

---

[176] Nous appelons ici **quasi-groupe additif canoniquement ordonné** $(B, +, \geq)$, un ensemble B muni :

(a) d'une loi de composition interne additive + commutative ($\forall\ (a, b) \in B^2$, $a + b = b + a$), associative (($\forall\ (a, b, c) \in B^3$, $(a + b) + c = a + (b + c)$), possédant un élément neutre 0_B, et

(b) d'une relation d'ordre total ≥ vérifiant : pour $(a, b) \in B^2$, $a \geq b$ si et seulement si il existe $c \in B$ tel que $a = b + c$.

Pour plus d'information sur les structures algébriques, voir par exemple REINHARDT & SOEDER [1997] ou MINOUX [1998].

[177] La courbe de Lorenz d'une distribution x de ressources entre n individus, a été définie sur le pourcentage d'individus $L_x(i / n)$. Mais pour simplifier l'écriture, nous noterons dans cette section directement $L_x(i)$. Cette simplification n'entraîne aucun biais étant donné que le nombre d'individus n est supposé constant.

$$S(c_{GL}[\text{incomparable}, x], V, D) = \mathcal{D}_V \setminus (S(c_{GL}[\text{surclasse}, x], V, D) \cup S(c_{GL}[\text{sousclasse}, x], V, D))$$

Nous nous intéressons ici à la conception d'algorithmes de filtrage pour les contraintes globales des deux contraintes souples $\{c_L[\alpha, z]\}_{\alpha, z}$ et $\{c_{GL}[\alpha, z]\}_{\alpha, z}$.

Tout d'abord, décrivons quelques relations entre les contraintes globales de Lorenz et les contraintes globales de Lorenz généralisées faisant intervenir les attitudes fondamentales :

Propriétés 7 : Étant donné un triplet (V, D, x) composé d'un ensemble de variables V, un ensemble de domaines D, à raison d'un $D(v)$ par variable v de V, les domaines étant inclus dans $({}^{+}, +, \geq)$ ou $(\ , +, \geq)$, et un élément x du produit cartésien des domaines de D, nous avons :

$$S(c_L[\text{indifférent}, x], V, D) = S(c_{GL}[\text{indifférent}, x], V, D) ;$$
$$S(c_L[\alpha, x], V, D) \subseteq S(c_{GL}[\alpha, x], V, D), \text{ pour } \alpha \in \{\text{mieux, pire}\} ;$$
$$S(c_L[\text{incomparable}, x], V, D) \supseteq S(c_{GL}[\text{incomparable}, x], V, D).$$

Ces propriétés sont des conséquences directes de la définition des deux dominances. A partir des Propriétés 7 et des Propriétés 3 (§ 5.2.4.1), les relations entre les attitudes composées peuvent être déduites.

Nous avons vu au § 7.2.2 que certaines contraintes de Lorenz possèdent une description linéaire connue, pouvant donc être utilisée pour établir un niveau d'inverse consistance (la LIB-consistance). Les contraintes proposent une alternative que nous allons développer ici.

Dans le reste de cette section nous allons surtout nous intéresser aux attitudes "indifférent" et "surclasse". Ainsi, au § 7.4.3.2 nous définissons une transformation polynomiale de la contrainte globale en l'indifférence vers la contrainte de cardinalité généralisée GCC, permettant ainsi d'utiliser directement les algorithmes de filtrage de cette contrainte dure. Ensuite (§ 7.4.3.3), nous présentons un petit exemple basé sur le problème de pure répartition. Aux § 7.4.3.4 et 7.4.3.5 nous explicitons les relations entre les contraintes globales de Lorenz d'un coté, et les contraintes de somme et de tri de l'autre coté.

7.4.3.2. Filtrage Global pour l'Indifférence par les Contraintes GCC et ALLDIFF

Un algorithme de filtrage pour la consistance inverse globale peut être obtenu pour la contrainte souple de Lorenz en l'indifférence et en une solution x, en utilisant la contrainte de cardinalité généralisée GCC (Annexe A1). Nous avons le résultat suivant :

Proposition 21 : Il existe une transformation polynomiale de la contrainte de Lorenz en (indifférent, x) vers la contrainte de cardinalité généralisée $c_{GCC}(V, D_x, B_x, [l, u])$.

Preuve : Étant donné un ensemble de variables V, un ensemble de domaines D, à raison d'un $D(v)$ par variable v de V, les domaines étant inclus dans $({}^{+}, +, \geq)$ ou $(\ , +, \geq)$, et un élément x du produit cartésien $\mathcal{D}_V$ des domaines de D, l'instance de GCC est déduit comme suit :

B_x les différentes valeurs des composantes de x : $B_x = \{x(v) \text{ tel que } v \in V\}$;

D_x les domaines discrets en extension obtenus à partir de D, en prenant les valeurs communes avec B_x : $D_x(v) = D(v) \cap B_x$ pour tout $v \in V$;

et pour finir,

l et u les deux tuples de $\text{card}(B_x)$ composantes, associant à chaque valeur de B_x, le nombre de fois qu'elle apparaît dans la distribution x :

$$l(d) = u(d) = \text{card}(\{v \in V \text{ tel que } x(v) = d\}) \text{ pour tout } d \in B_x.$$

Maintenant montrons l'égalité entre les deux sémantiques. Notons $\mathcal{D}_{V,x}$ le produit cartésien des domaines de D_x. La sémantique de la contrainte GCC en cette instance s'écrit :

$$S(c_{\text{GCC}}(V, D_x, B_x, [l,u])) = \left\{ \begin{array}{l} y \in \mathcal{D}_{V,x} \text{ tel que : pour tout } d \in B_x, \\ \qquad l(d) \le \text{card}(\{v \in V \text{ tel que : } y(v) = d\}) \le u(d) \end{array} \right\}$$

or, par construction $x \in S(c_{\text{GCC}}(V, D_x, B_x, [l, u]))$ et $l(d) = u(d)$. Donc :

$$S(c_{\text{GCC}}(V, D_x, B_x, [l,u])) = \left\{ \begin{array}{l} y \in \mathcal{D}_{V,x} \text{ tel que : pour tout } d \in B_x, \\ \text{card}(\{v \in V \text{ tel que : } y(v) = d\}) = \text{card}(\{v \in V \text{ tel que : } x(v) = d\}) \end{array} \right\}$$

Il est alors très facile de construire deux numérotations l_x et l_y des éléments de V, l'une dépendant de x l'autre de y respectant les conditions suivantes : $z(v1) > z(v_2) \Rightarrow l_z(v_1) \ge l_z(v_1)$ avec $z \in \{x, y\}$. Nous obtenons alors :

$$S(c_{\text{GCC}}(V, D_x, B_x, [l,u])) = \left\{ y \in \mathcal{D}_{V,x} \text{ tel que : pour tout } i \in \{1, \ldots, \text{card}(V)\},\ y\left(l_y^{-1}(i)\right) = x\left(l_x^{-1}(i)\right) \right\}$$
$$= S(c_{\text{L}}[\text{indifférent}, x], V, D) \qquad \square$$

Remarquons que lorsque les composantes de x sont deux à deux différentes, un algorithme de filtrage de la contrainte ALLDIFF[178] en (V, D_x) peut être utilisé pour $c_{\text{L}}[\text{indifférent}, x]$. Dans ce cas, $S(c_{\text{ALLDIFF}}(V, D_x)) = \{y \in \mathcal{D}_{V,x}$ tel que $y(v_1) \ne y(v_2)$ pour tout $v_1, v_2 \in V$ et $v_1 \ne v_2\}$
$= S(c_{\text{L}}[\text{indifférent}, x], V, D)$.

7.4.3.3. Résolution Graphique sur le Simplexe de Dimension 3 pour le Surclassement

Présentons un exemple simple d'allocation de ressources (problème de pure répartition à domaines contraints individuellement) de trois individus $E = \{e_1, e_2, e_3\}$ associés aux variables $V = \{v_1, v_2, v_3\}$ indiquant chacune la quantité de ressources à attribuer à l'individu associé. Leurs domaines sont donnés ci-dessous :

$D(v_1) = [5 ; 15] \cup [33 ; 90]$; $D(v_2) = [5 ; 10] \cup [25 ; 80]$; $D(v_3) = [5 ; 80]$

Quelles sont les valeurs de domaines pouvant être supprimées par la contrainte de Lorenz en (surclasse, x) ?

Notons $Q = x(v_1) + x(v_2) + x(v_3) = 100$ la quantité totale de ressources à distribuer. La solution idéale est $x^* = (100 / 3, 100 / 3, 100 / 3)$. La Figure 48 représente cet exemple sur le triangle de Kolm.

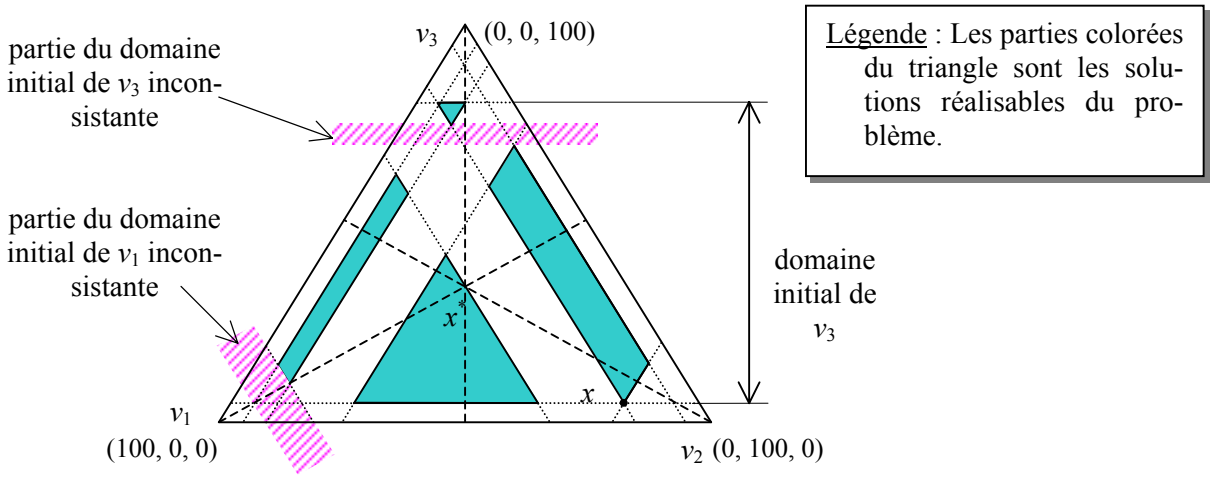


Figure 48 : Résolution Graphique de la contrainte de Lorenz en (surclasse, x) pour le problème de pure répartition sur 3 individus et une quantité totale de ressource égale à 100.

Une procédure graphique consiste à :

[178] Voir Annexe A1.

(1) revoir les bornes des domaines en fonction des valeurs minimales et maximales des composantes de la solution x : $D(v) \subseteq [\min_{u \in V}\{x(u)\}, \max_{u \in V}\{x(u)\}]$.

Pour notre exemple, la borne supérieure de $D(v_1)$ est baissée à 80.

(2) mettre à jour chaque domaine $D(v_i)$ en (a) effectuant la somme des domaines du couple de variables $V \setminus \{v_i\}$, puis en (b) soustrayant de la quantité Q le nouveau domaine obtenu. Le nouveau domaine de v_i correspond alors à l'intersection de ce domaine et du domaine initial de v_i.

Pour notre exemple, seul $i = 3$ effectue un élagage. Son itération est présentée ci-dessous :

$D(v_1) + D(v_2)$	[5 ; 10]	[25 ; 80]
[5 ; 15]	[10 ; 25]	[30 ; 95]
[33 ; 80]	[38 ; 90]	[58 ; 160]

$\Rightarrow S = [10 ; 25] \cup [30 ; 160]$. Ainsi,
$\{Q\} \setminus S = [-60 ; 70] \cup [75 ; 90]$, et par conséquent,
$D(v_3) = [5 ; 70] \cup [75 ; 80]$

Dans le cas général, qu'est-il possible de faire ? Pouvons-nous envisager l'automatisation de la procédure précédente ? Voyons les difficultés que cela entraîne au prochain paragraphe.

7.4.3.4. La Contrainte de Somme et la Contrainte Souple de Lorenz

La procédure précédente effectue entre autre une somme d'ensembles, laquelle a été très étudiée en arithmétique des intervalles, et connue sous le nom de contrainte de somme en programmation par contraintes.

❖ La Contrainte de Somme

La **contrainte de somme** $c_{\text{SUM}}(V, w, D)$ considère en entrée :

$V = \{v_1, \ldots, v_n\}$ un ensemble de variables non vide V en extension :.

w une variable supplémentaire, et

D un ensemble de domaines associant à chaque variable v de $V \cup \{w\}$ un domaine $D(v)$ inclus dans un demi-groupe additif $(H, +)$ (comme l'ensemble des entiers naturels $(\ \ , +)$).

Notons $\mathcal{D}_V = D(v_1) \times \ldots \times D(v_n)$ le produit cartésien des domaines de V. Les solutions réalisables de la contrainte de somme sont les éléments de $\mathcal{D}_V \times D(w)$ dont la somme des n premières composantes est égale à la dernière composante :

$$S(c_{\text{SUM}}(V, w, D)) = \{y \in \mathcal{D}_V \times D(w) \text{ tel que } y(v_1) + \ldots + y(v_n) = y(w)\}.$$

Très étudiée notamment en arithmétique sur les intervalles, cette contrainte n'a connu un intérêt pour les domaines conçus comme une union finie d'intervalles, que ces dernières années avec les travaux d'HICKEY [2001] et HICKEY *et al.* [2001].

Lorsque les domaines sont des intervalles fermés, nous connaissons des algorithmes de filtrage établissant la consistance inverse globale (voir par exemple JAULIN *et al.* [2001], HICKEY *et al.* [2001]). Lorsque les intervalles sont en extension, nous connaissons un algorithme établissant la borne-consistance (RÉGIN & RUEHER [2000]). Mais, existe-t-il un algorithme polynomial en la taille des domaines résolvant le problème d'inverse consistance globale lorsque les domaines sont donnés en extension ? Qu'en est-il des domaines codés en unions finis d'intervalles disjoints ? Les intervalles fermés sont stables pour l'addition et la taille de la sortie reste polynomiale en la taille de l'entrée. Mais sorti de ce cadre, des problèmes de complexité peuvent apparaître. La remarque suivante illustre cette affirmation avec les unions finis d'intervalles disjoints.

Remarque 4 : La somme d'ensembles formés d'une union finie d'intervalles disjoints peut retourner un domaine formé d'une union finie d'intervalles disjoints de taille exponentielle en la taille des données :

Prenons l'exemple (cas pathologique) d'un ensemble D de domaines sur n variables V, formés de la manière suivante :

$D(v_1) = [1, 1 + \varepsilon] \cup [2, 2 + \varepsilon] \cup \ldots \cup [m, m + \varepsilon]$,
$D(v_2) = [0, \varepsilon] \cup [m, m + \varepsilon] \cup [2m, 2m + \varepsilon] \cup \ldots \cup [m^2 - m, m^2 - m + \varepsilon]$,
...
$D(v_n) = [0, \varepsilon] \cup [m^{n-1}, m^{n-1} + \varepsilon] \cup [2m^{n-1}, 2m^{n-1} + \varepsilon] \cup \ldots$
$\cup [m^n - m^{n-1}, m^n - m^{n-1} + \varepsilon]$.

En prenant $0 \leq \varepsilon < 1 / n$ nous obtenons :

Chaque domaine est formé de m intervalles, et la somme d'un tel système produit un ensemble formé de m^n intervalles disjoints. Ainsi, au départ, nous avions $n.m$ intervalles, et à la fin nous arrivons à un nombre exponentiel d'intervalles. En prenant initialement le domaine $D(w) = [1, m^n]$ pour la variable de somme w, le nombre d'intervalles reste inchangé : exponentiel en n.

En ce qui concerne la taille des domaines, chaque borne d'intervalle est codée dans une base b > 1 impliquant une taille maximale en $O(n \cdot \log_b(m))$ pour les bornes du domaine initial, taille qui est conservée par les bornes du domaine résultat.

En définitive, nous passons d'un domaine de taille en $O(n^2 \cdot m \cdot \log(m))$ en un domaine de taille en $O(n \cdot m^n \cdot \log(m))$ exponentiel en n.

En résumé, pour un codage usuel donné, la consistance inverse globale peut conduire pour la contrainte de somme à des domaines de taille exponentielle en la taille des données de départ. Dans la terminologie de la théorie de la complexité, la somme est un problème sortie-polynomial en temps[179]. Pour traiter ce type de problèmes, des algorithmes polynomiaux en la taille de l'entrée et de la sortie peuvent être conçus. D'autres solutions sont possibles comme : l'utilisation d'un niveau de consistance plus faible conservant la polynomialité de la taille de la sortie en fonction de celle de l'entrée, ou encore, l'utilisation d'un codage des domaines et contraintes approprié comme ceux basés sur la discrétisation d'ensembles (SAM-HAROUD & FALTINGS [1996], JAULIN *et al*. [2001] parmi d'autres).

❖ Contrainte de Somme pour le Filtrage de la Contrainte Souple de Lorenz

Comme nous avons vu avec l'exemple du § 7.4.3.3, la contrainte de Lorenz en (surclasse, x) peut utiliser la somme pour établir un niveau de consistance. La proposition suivante permet d'éclaircir cette dépendance :

Proposition 22 : Étant donné un quadruplet (V, D, α, x) composé d'un ensemble de variables V, un ensemble de domaines D, à raison d'un $D(v)$ par variable v de V, les domaines étant inclus dans $(\ ^+, +, \geq)$ ou $(\ , +, \geq)$, α une attitude incluse dans l'ensemble {mieux, pire, indifférent}, et un élément x du produit cartésien des domaines de D, nous avons :

$$S(c_L[\alpha, x], V, D) \subseteq \text{proj}_V(S(c_{SUM}(V, w_x, D_x))), \qquad (7\text{–}41)$$

avec w_x une variable associée à x, son domaine étant égal au singleton $\{Q_x\}$ avec $Q_x = \sum_{v \in V} x(v)$, et D_x les domaines associés à $V \cup \{w_x\}$ ($\Leftrightarrow D_x = D \cup \{\{Q_x\}\}$).

Preuve : Elle est directe : Seules les solutions y dont la somme des composantes est égale à celle de x lui sont comparables. ❑

En conclusion, l'utilisation de la contrainte de somme permet d'éliminer certaines valeurs de domaines inconsistantes avec les contraintes de Lorenz en $\alpha \subseteq$ {mieux, pire, indifférent}. Par conséquent, si l'attitude α contient l'attitude fondamentale incomparable, alors toute distribution dont la somme des composantes est différente de celle de x est réalisable :

[179] Pour des références, voir fin du § 2.1.1.2.

$$\left\{ y \in \mathcal{D}_V \text{ tel que : } \sum_{v \in V} y(v) \neq \sum_{v \in V} x(v) \right\} \subseteq S(c_{\mathrm{L}}[\text{incomparable}, x], V, D)$$

Pour la contrainte souple de Lorenz généralisée, seule l'indifférence vérifie l'inclusion (7–41) (voir l'égalité parmi les Propriétés 7).

Nous allons considérer dans le reste de cette sous-section, et dans la sous-section suivante (seconde partie du § 7.4.3.5) le cadre des problèmes d'allocation de ressources. Nous appellerons donc, instance générale de problème d'allocation de ressources le quadruplet $I_{AR} = (E, V, D, D_r)$ formé d'une population E, un ensemble de variables V et de domaines D, à raison d'une variable $v(i)$ et d'un domaine $D(v(i)) = D(i)$ par individu i, $D(i)$ correspondant aux différentes quantités pouvant être attribuées à l'individu i ; et D_r les variations possibles pour la quantité totale de ressources. Nous avons : $\text{card}(E) = \text{card}(V) = \text{card}(D) > 1$. Étudions une contrainte formée d'une combinaison de la contrainte de GL en (α, x) et des contraintes de borne sur la quantité totale de ressources. Nous l'appelons contrainte de Lorenz généralisée en (α, x) pour l'instance I_{AR}, et notée $c_{\mathrm{GL}}[\alpha, x](I_{AR})$. Son domaine réalisable est défini comme suit :

$$S(c_{\mathrm{GL}}[\alpha, x](I_{AR}), V, D) = \left\{ y \in \mathcal{D}_V \text{ tel que : } y \ \alpha_{\mathrm{GL}} \ x, \text{ et } \sum_{v \in V} y(v) \in D_r \right\} \qquad (7\text{–}42)$$

Dans un tel cadre, il est possible d'effectuer un élagage des domaines avec la contrainte de somme, pour certaines contraintes globales de Lorenz généralisées. Nous avons ainsi la proposition suivante :

Proposition 23 : Étant donné un triplet (I_{AR}, α, x) composé d'une instance générale d'allocation de ressources $I_{AR} = (E, V, D, D_r)$, d'une attitude α incluse dans l'ensemble {mieux, pire, indifférent}, et d'une distribution de ressources x pour E, alors :

$$S(c_{\mathrm{GL}}[\alpha, x](I_{AR}), V, D) \subseteq \text{proj}_V(\ S(c_{\mathrm{SUM}}(V, u_r, D_1))\ ), \qquad (7\text{–}43)$$

avec u_r une variable associée à la quantité totale de ressources, son domaine étant égal à D_r, et D_1 les domaines associés à $V \cup \{u_r\}$ ($\Leftrightarrow D_1 = D \cup \{D_r\}$).

Preuve : Elle aussi est directe. ❑

Continuons à étudier les relations entre les contraintes de Lorenz et l'instance générale de problème d'allocation de ressources.

❖ La Contrainte de Lorenz Généralisée Ordonnée en le Surclassement

Considérons une instance générale de problème d'allocation de ressources $I_{AR} = (E, V, D, D_r)$ formée d'une population E, d'un ensemble de variables V et de domaines D, à raison d'une variable $v(i)$ et d'un domaine $D(v(i)) = D(i)$ par individu i, $D(i)$ correspondant aux différentes quantités pouvant être affectées à l'individu i ; et D_r les variations possibles pour la quantité totale de ressources. A partir de I_{AR} et d'une solution initiale x – réalisable ou non –, étudions une version combinée avec d'autres contraintes, de la contrainte de Lorenz généralisée en (surclasse, x) appelée OGLO pour *ordered generalized Lorenz constraint in outranking* : $c_{\mathrm{OGLO}}(V, u_r, D, D_r, x)$. Elle considère en entrée, en plus des informations de l'instance générale I_{AR}, une variable u_r associée à la quantité totale de ressources, et la distribution x. La sémantique de cette contrainte est :

$$S(c_{\mathrm{OGLO}}(V, u_r, D, D_r, x)) = \left\{ (y_1, y_2) \in \mathcal{D}_V \times D_r \text{ tel que : } \left| \begin{array}{l} y_1 \succcurlyeq_{\mathrm{GL}} x \\ y(v_k) \leq y(v_{k+1}), \text{ pour tout } 1 \leq k < n \\ L_{y_1}(n) = y_2 \end{array} \right. \right\}$$

avec n = card(V). La contrainte OGLO correspond à la combinaison de la contrainte de Lorenz généralisée en (surclasse, x), avec $n - 1$ contraintes de rangement par ordre croissant des domaines, et avec une contrainte de bornes sur la quantité totale de ressources. Nous allons présenter ici un algorithme de borne-consistance pour la contrainte OGLO.

Tout d'abord, les $n - 1$ contraintes de rangement par ordre croissant imposent que les domaines D soient normalisés. Un ensemble de domaines est **normalisé**[180] si $\min(D(v_k)) \leq \min(D(v_{k+1}))$ et $\max(D(v_k)) \leq \max(D(v_{k+1}))$ pour tout $1 \leq k < n$. La normalisation s'obtient algorithmiquement en appliquant les deux boucles suivantes :

pour ($k = 1$ **à** $n - 1$) **faire**
  $\min(D(v_{k+1})) \leftarrow \max\{\min(D(v_k)), \min(D(v_{k+1}))\}$
fin pour
pour ($k = n$ **à** 2) **faire**
  $\max(D(v_{k-1})) \leftarrow \min\{\max(D(v_k)), \max(D(v_{k-1}))\}$
fin pour

Un premier élagage peut ainsi être fait avec la normalisation.

Étudions maintenant les suppressions de valeurs intervenant dans le cas particulier où $D(v) = [0, \max(D_r)]$ pour tout v de V. Rappelons d'abord ce que signifie l'expression : "la distribution y surclasse la distribution x selon la dominance de Lorenz généralisée" :

$$y \succcurlyeq_{\mathrm{GL}} x \;\Leftrightarrow\; [\, L_y(k) \geq L_x(k) \;\; \text{pour tout } k \in \{1, \ldots, n\} \,]$$
$$\Leftrightarrow \left[\sum_{i=1}^{k} y(v_{\sigma_y(i)}) \geq \sum_{i=1}^{k} x(v_{\sigma_x(i)}) \;\; \text{pour tout } k \in \{1, \ldots, n\} \right]$$

σ_x et σ_y étant deux permutations de n éléments rangeant les composantes de respectivement x et y par ordre croissant. Rappelons que $z(v_{\sigma_z(k)})$ est la quantité attribuée à la personne de rang k par la distribution z, et que $L_z(k)$ est la courbe de Lorenz discrète de la distribution z en k.
Si $\max(D_r)$ est inférieur à la quantité totale distribuée par la distribution x, alors la contrainte est inconsistante. Dans le cas contraire, les seules valeurs de D_r consistantes avec la contrainte OGLO vérifient : $\min(D_r) \geq L_x(n)$. De plus, les bornes entre lesquelles se trouve la quantité qui sera attribuée à l'individu de rang k pour tout $1 \leq k \leq n$ vérifient les conditions suivantes :

$$\min(D(v_k)) = \begin{cases} \dfrac{1}{k} L_x(k) & \text{pour } 1 \leq k < n \\ \dfrac{1}{n} \max\{L_x(n), \min(D_r)\} & \text{pour } k = n \end{cases} \tag{7–44}$$

$$\max(D(v_k)) = \begin{cases} \dfrac{1}{n-k+1} (\max(D_r) - L_x(k-1)) & \text{pour } 2 \leq k \leq n \\ \dfrac{1}{n} \max(D_r) & \text{pour } k = 1 \end{cases} \tag{7–45}$$

Donnons quelques explications pour la borne inférieure du domaine $D(v_k)$: La quantité de ressources $y(v_k)$ attribuée à l'individu de rang k est minimale, pour $k < n$, si $L_y(k) = L_x(k)$ et que les quantités de ressources attribuées aux individus moins bien lotis que k est égale à la quantité attribuée à l'individu de rang k. En d'autres termes, la distribution entre les individus moins bien lotis que l'individu de rang $k + 1$ est équitable. Ainsi, ce minimum vaut $(1 / k) \cdot L_x(k)$. Dans le cas de l'individu le mieux loti, $k = n$, deux possibilités se présentent : soit la quantité totale $L_x(n)$ distribuée par x est inférieure à $\min(D_r)$, soit elle est supérieure ou égale. Pour la première possibilité, la quantité minimale qui peut lui être attribuée, et qui reste réali-

[180] Pour une notion équivalente, mais dans le cadre de la contrainte de Tri que nous verrons au § 7.4.3.5, voir BLEUZEN-GUERNALEC & COLMERAUER [2000] et MEHLHORN & THIEL [2000].

sable est la distribution équitable de la quantité min(D_r) entre les n individus. Pour la seconde possibilité, il suffit de répartir équitablement la quantité totale distribuée par x.
Expliquons maintenant la borne supérieure de $D(v_k)$: La quantité de ressources affectée à l'individu de rang $k > 1$ est maximale lorsque les $n - k + 1$ individus les mieux lotis se partagent le maximum de ressources, laissant aux $k - 1$ autres juste assez pour ne pas être en dessous de la distribution x. Ainsi, la quantité maximale $Q_{max}(k)$ pouvant être distribuée aux individus de rang supérieur ou égal à k est égale à la quantité totale maximale de ressources pouvant être distribuée à E (max(D_r) en l'occurrence), à laquelle sont soustraites les quantités minimales pouvant être distribuées aux individus de rang inférieur à k (en d'autres termes $L_x(k - 1)$). Ensuite, sachant que l'individu de rang k ne peut pas avoir plus qu'un individu de rang supérieur, la quantité maximale qui peut être affectée à k est la part d'un individu de la distribution équitable de la quantité $Q_{max}(k)$ entre k individus. Lorsque $k = 1$, l'individu le plus mal loti reçoit une quantité de ressources maximale, lorsque toutes les ressources sont distribuées équitablement.

Remarquons que lorsque les ressources sont indivisibles (D_r est discret), certaines précautions sont à prendre. Par exemple, si 46 unités de ressources doivent être distribuées équitablement entre 4 personnes, les personnes de rang 1 et 2 recevront chacune 11 unités et les deux autres 12 unités de manière à ce que la somme fasse 46, que les contraintes de rangs soient respectées et que la distribution choisie épouse au mieux (soit la moins éloignée de) la distribution équitable mais irréalisable $\frac{23}{2}(1, 1, 1, 1)$. La même méthodologie d'arrondi peut aussi être appliquée dans le cas plus général évoqué à la note en bas de page numéro 176.

Après ces explications, une preuve par l'absurde simple permet de prouver ces égalités. Nous ne la donnons pas ici, et laissons le soin au lecteur de la faire.

Qu'en est-il dans le cas général ? L'algorithme suivant permet d'établir un niveau de consistance inférieur ou égal à la borne-consistance pour la contrainte OGLO. Étant donnée une instance de contrainte OGLO (V, u_r, D, D_r, x), l'algorithme se déroule en 5 étapes :

(1) Normaliser les bornes des domaines de D selon le rang ;

(2) Modifier les bornes inférieures et supérieures des domaines de D selon la distribution x (formules (7–44) et (7–45)) ;

(3) Calculer les bornes de D_r sachant que :

$$\min(D_r) \geq \max\left\{L_x(n), \sum_{i=1}^{n} \min(D(v_k))\right\} \qquad \text{et} \qquad \max(D_r) \leq \sum_{i=1}^{n} \max(D(v_k)) \ ;$$

(4) Ajuster les bornes supérieures des domaines de D sachant que :

$$\max(D(v_k)) \leq \begin{cases} \frac{1}{n}\max(D_r) & \text{pour } k = 1 \\ \frac{1}{n-k+1}\left(\max(D_r) - \sum_{i=1}^{k-1} \min(D(v_i))\right) & \text{pour } 2 \leq k \leq n \end{cases}$$

(5) Réajuster la borne supérieure de D_r si nécessaire.

La troisième étape est justifiée par les conditions de borne-consistance de la contrainte de somme sous-jacente (voir RÉGIN & RUEHER [2000, proposition 1]). Et la quatrième étape est justifiée par le fait que la quantité maximale pouvant être allouée à l'individu de rang k est obtenue en distribuant le strict minimum aux $k - 1$ individus les moins bien lotis, et en répartissant équitablement la quantité totale maximale de ressources restante aux $n - k + 1$ individus les mieux lotis.

Cet algorithme est valable dans le cas général. Mais il peut être simplifié s'il est utilisé pour un maintien de consistance, selon que seules les bornes de domaines ont été modifiées, ou que la solution x est inchangée, etc.

Continuons notre comparaison de la contrainte souple de Lorenz et de celle de Lorenz généralisée avec des contraintes globales dures connues.

7.4.3.5. Contrainte de Tri et Contrainte Souple de Lorenz en le Surclassement

La contrainte de Lorenz généralisée en (surclasse, x) semble très proche de la contrainte de tri. Nous la présentons ci-dessous :

❖ La Contrainte de Tri

La **contrainte de tri** $c_{\text{SORT}}(V, W, D)$ considère en entrée :

V et W deux ensembles de variables en extension, non vides et de même cardinal :
$V = \{v_1, \ldots, v_n\}$ et $W = \{w_1, \ldots, w_n\}$ avec $\text{card}(V) = \text{card}(W) = n > 0$; et

D un ensemble de $2n$ domaines à raison d'un domaine $D(u)$ par variable u dans $V \cup W$; chaque domaine étant un sous-ensemble d'un même ensemble non vide $(H, \geq)$ totalement ordonné : $D(u) \subseteq H$.

Les solutions réalisables de la contrainte de tri sont les éléments de $\mathcal{D}_{V \cup W} = D(v_1) \times \ldots \times D(v_n) \times D(w_1) \times \ldots \times D(w_n)$ dont les n dernières composantes sont obtenues en triant les n premières composantes :

$$S(c_{\text{SORT}}(V,W,D)) = \left\{ \begin{array}{l} y \in \mathcal{D}_{V \cup W} \text{ tel que :} \\ \quad \left| \begin{array}{l} y(w_1) \geq \cdots \geq y(w_n) \text{ et que} \\ \exists\, \sigma_y \text{ une permutation de } \{1,\ldots,n\} \text{ vérifiant } y(w_i) = y(v_{\sigma_y(i)}) \end{array} \right. \end{array} \right\}.$$

Cette contrainte a été présentée pour la première fois par Older *et al.* (1995)[181] comme moyen de modélisation et résolution de problèmes d'ordonnancement d'ateliers. Zhou [1996] et [1997] l'a utilisée par la suite pour résoudre des problèmes de job-shop difficiles (sa contrainte de tri se présente sous une forme légèrement différente : L'ensemble des solutions réalisables est définit par les $3n$-uplets $(x, \sigma_x) \in S(c_{\text{SORT}}(V, W, D)) \times \mathcal{S}_{2n}$, avec $\mathcal{S}_{2n}$ l'ensemble des permutations de $2n$ éléments). BLEUZEN-GUERNALEC & COLMERAUER [1997] et [2000] ont présenté un algorithme en $O(n \cdot \log(n))$ résolvant le problème de borne-consistance sur les intervalles fermés ($\Leftrightarrow D(u) = [a, b] \subseteq H$). Par la suite, MEHLHORN & THIEL [2000] ont apporté des améliorations pratiques à l'algorithme – en se ramenant à un problème de couplage dans les graphes bipartis – et ont présenté des versions spécifiques pour la contrainte ALLDIFF et la contrainte de Permutation.

La Figure 49 donne un exemple, extrait de BLEUZEN-GUERNALEC & COLMERAUER [1997], de l'application de la contrainte de tri.

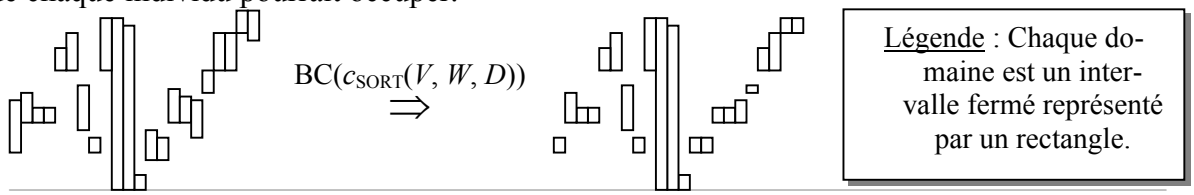
❖ Contrainte de Tri pour un Filtrage de la Contrainte de Lorenz Généralisée en le Surclassement

Considérons une instance de problème général d'allocation de ressources $I_{AR} = (E, V, D, D_r)$. Intéressons-nous ici à concevoir un algorithme d'inverse consistance pour la contrainte de Lorenz généralisée en (surclasse, x) pour l'instance I_{AR} : $c_{\text{GL}}[\text{surclasse}, x](I_{AR})$ (voir formule (7–42)). Nous recherchons les domaines $\text{GIC}(c_{\text{GL}}[\text{surclasse}, x](I_{AR}), V, D)(v)$, pour tout $v \in V$, ou une approximation.

La contrainte de tri et la contrainte OGLO semblent être complémentaires pour la contrainte de Lorenz généralisée en (surclasse, x) pour l'instance I_{AR}. Intuitivement, la contrainte

[181] OLDER William J., SWINKELS G. M. & VAN EMDEN M. H. [1995] – Getting to the Real Problem : Experience with BNR Prolog in OR – *in Proceedings of the 3rd International Conference on the Practical Applications of Prolog* (*PAP'95*), Alinmead Software Ltd, ISBN 0 9 525554 0 9, avril 1995, Paris.

de Lorenz pourrait être décomposée en deux contraintes partielles : la contrainte OGLO serait utilisée pour identifier les valeurs pouvant être attribuées à l'individu de rang k en fonction de la solution x et des valeurs possibles D_r pour la quantité totale de ressources. La contrainte de tri identifierait les compatibilités de domaines entre les quantités pouvant être attribuées à chaque individu, et celles pouvant être attribuées à chaque rang. La contrainte de tri supprimera les valeurs de domaines qui aboutiraient forcément à une incohérence en fonction du rang que chaque individu pourrait occuper.



Instance :	Résultat :
$n = 11$; $(H, \geq) = (\quad, \geq)$ $\mathcal{D}_{V \cup W} = [10, 24] \times [18, 26] \times [18, 22] \times [18, 22] \times [32, 38] \times [32, 42] \times [16, 28] \times [10, 14] \times [32, 46] \times [0, 46] \times [0, 44] \times [0, 4] \times [8, 16] \times [8, 14] \times [18, 27] \times [18, 25] \times [14, 24] \times [26, 32] \times [32, 42] \times [32, 42] \times [42, 48] \times [40, 48]$.	$\mathcal{D}'_{V \cup W} = [10, \mathbf{14}] \times [18, 26] \times [18, 22] \times [18, 22] \times [32, 38] \times [32, 42] \times [16, 28] \times [10, 14] \times [32, 46] \times [0, 46] \times [0, 44] \times [0, 4] \times [\mathbf{10}, \mathbf{14}] \times [\mathbf{10}, 14] \times [18, \mathbf{22}] \times [\mathbf{18}, 22] \times [18, 24] \times [26, \mathbf{28}] \times [32, \mathbf{38}] \times [32, 42] \times [42, \mathbf{46}] \times [\mathbf{42}, \mathbf{46}]$.

Figure 49 : Schéma de visualisation de l'établissement de la borne-consistance pour la contrainte de tri.

La proposition suivante permet de confirmer cette intuition :

Proposition 24 : Étant donnée une instance générale de problème d'allocation de ressources (E, V, D, D_r), et un élément x du produit cartésien des domaines de D, alors :

$$S(c_{\text{GL}}[\text{surclasse}, x](I_{AR}), V, D) = \text{proj}_V(\, S_{\text{CSP}}(\mathcal{R}(I_{AR}, x))\,)$$

avec :

$$\mathcal{R}(I_{AR}, x) = (V \cup W \cup \{u_r\}, D_{V \cup W} \cup \{D_r\}, \{c_{\text{TRI}}(V, W, D_{V \cup W}), c_{\text{OGLO}}(W, u_r, D_W, D_r, x)\})$$

C'est-à-dire, en notant $n = \text{card}(E)$, que $\mathcal{R}(I_{AR}, x)$ est un système de contraintes formé de :

(a) $2n + 1$ variables, sachant que V est l'ensemble des variables associé à la population E, W un ensemble de n variables associé aux n rangs possibles, u_r la variable associée à la quantité totale de ressources ;

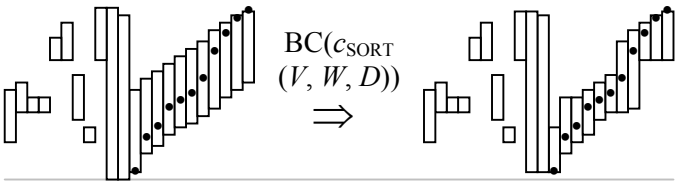
(b) $2n + 1$ domaine avec $D_{V \cup W} = D_V \cup D_W$ les domaines de V et de W, le domaine $D(w_k)$ de la variable w_k étant initialisé à $[0, \max(D_r)]$, si $\max(D_r) < +\infty$, et $[0, +\infty[$ sinon ;

(c) 2 contraintes, l'une de tri, l'autre OGLO, reliées par les variables W.

Preuve : Nous devons prouver que toute distribution $y \in S(c_{\text{GL}}[\text{surclasse}, x](I_{AR}), V, D)$ est la projection d'une solution réalisable du système de contraintes $\mathcal{R}(I_{AR}, x)$ sur V, et réciproquement. Il suffit d'appliquer les définitions. Partons de la définition de la contrainte de Lorenz généralisée en (surclasse, x) pour l'instance I_{AR} : Considérons une permutation σ_y de n éléments, rangeant la distribution y par ordre croissant, et z_y la distribution de $\mathcal{D}_V$ rangeant y par ordre croissant ($\Leftrightarrow z_y(v_k) = y(v_{\sigma_y(k)})$ pour tout $1 \leq k \leq n$). Alors par définition, $y(v_k) \in D(v_k)$, $y \succcurlyeq_{\text{GL}} x$ et il existe un $Q_y \in D_r$ tel que $L_y(n) = Q_y$. Ce qui est équivalent à dire que $(z_y, Q_y) \in S(c_{\text{OGLO}}(W, u_r, D_W, D_r, x))$, et $(y, z_y) \in S(c_{\text{TRI}}(V, W, D_{V \cup W}))$. Ce qui est aussi équivalent à dire que $(y, z_y, Q_y) \in S_{\text{CSP}}(\mathcal{R}(I_{AR}, x))$. ❑

En conséquence, nous pouvons utiliser les deux contraintes OGLO et Tri avec un algorithme de propagation de contraintes, pour établir un niveau de consistance de bonne qualité pour la contrainte $c_{\mathrm{GL}}[\mathsf{surclasse}, x](I_{AR})$.
Remarquons que ces deux contraintes ont en commun les variables W, mais aussi les contraintes de rangement par ordre croissant. Par conséquent, les procédures communes (la normalisation des domaines) peuvent être exécutées une seule fois.
Lorsque les domaines ne sont pas bornés ($\Leftrightarrow D(v) = D_r = {}^{+}$), le niveau de consistance obtenu en utilisant les formules (7–44) et un algorithme de borne-consistance pour la contrainte de tri est égal au niveau de consistance établi par un algorithme de LIB-consistance (Annexe A2) en utilisant le multipermutoèdre (§ 7.2.2.1).

La Figure 50 illustre la relation entre la contrainte de Tri, la contrainte OGLO et la contrainte de Lorenz en (surclasse, x) avec un problème d'allocation de ressources dont les données sont : un ensemble de 11 variables, les domaines associées sont ceux de l'exemple précédent (Figure 49), la solution x = (14, 23, 21, 19, 34, 39, 27, 11, 45, 43, 2) et une quantité totale de ressources fixe Q = 278.



Légende : Chaque domaine est un intervalle fermé représenté par un rectangle. Les points à l'intérieur des intervalles représentent la solution x dont les composantes ont été rangées par ordre croissant.

Instance contrainte de Lorenz en (surclasse, x) : n = 11 ; $(D, +, \geq) = (\ \ , +, \geq)$; Q = 278 $\mathcal{D}_V$ = [10, 24] × [18, 26] × [18, 22] × [18, 22] × [32, 38] × [32, 42] × [16, 28] × [10, 14] × [32, 46] × [0, 46] × [0, 44] x = (14, 23, 21, 19, 34, 39, 27, 11, 45, 43, 2)	Résultats du calcul de bornes selon l'algorithme de filtrage de la contrainte OGLO présenté en fin de § 7.4.3.4 : $\mathcal{D}_W$ = [2, 25] × [7, 27] × [9, 29] × [12, 31] × [14, 33] × [15, 35] × [17, 37] × [19, 40] × [22, 42] × [24, 44] × [26, 45].
Résultats de la contrainte de Tri sur(V, W, D) : $\mathcal{D}'_{V \cup W}$ = [10, 24] × [18, 26] × [18, 22] × [18, 22] × [32, 38] × [32, 42] × [16, 28] × [10, 14] × [32, **45**] × [**2**, **45**] × [**2**, 44] × [2, **24**] × [7, 27] × [**10**, 29] × [12, 31] × [**16**, 33] × [**18**, 35] × [**18**, 37] × [19, 40] × [**32**, 42] × [**32**, 44] × [**32**, 45] Résultats de la contrainte de Lorenz en (surclasse, x) : $\mathcal{D}'_V$.	

Figure 50: Exemple d'utilisation de la contrainte de tri et OGLO, pour établir un niveau de consistance pour la contrainte de Lorenz en (surclasse, x).

Remarquons que la contrainte de tri donne la possibilité de contraindre les quantités possibles de chaque individu indirectement selon le rang qu'ils occupent.

7.4.4. Règle de Pareto comme Outil d'Agrégation des Préférences Individuelles sur les Positions

Nous avons vu à la fin du § 4.2.1.2 qu'il est possible de faire usage de règles d'agrégations simples sur l'ensemble des positions pour agréger l'information préférentielle étendue en une seule relation binaire préférentielle étendue. Nous avons ensuite proposé au § 6.4 d'utiliser la règle d'agrégation de Pareto pour agréger l'information préférentielle de type sympathie étendue de notre problème d'élaboration de planning. Nous allons voir ici, comment cette règle de Pareto peut être mise en œuvre.

7.4.4.1. Règle de Pareto et PB-Contrainte Souple Étendue

La règle de Pareto sur $\mathcal{D}_V \times I_e$ correspond littéralement à la contrainte souple $\{c_P[\alpha, (x, i)]\}_{\alpha, (x, i)}$. Elle correspond aussi à une relation binaire de préférence étendue notée $\{c_P[\alpha, x, i, j]\}_{(\alpha, x, i, j)}$ dont la définition est la suivante :

$S(c_P[\alpha, x, i, j], V, D) = \{y \in \mathcal{D}_V$ tel que : $(y, j) \in S(c_P[\alpha, (x, i)], D \times I_e)\}$

Nous allons écrire $S(c_P[\alpha, x, i, j], V, D)$ directement à partir du profil de sympathie étendu de préférences individuelles $\{c_k[\alpha, x, i, j]\}_{(\alpha, x, i, j)}$ pour tout $k \in I_{se}$, sans passer par $S(c_P[\alpha, (x, i)], D \times I_e)$).

$S(c_P[\alpha, x, i, j], V, D) = \{y \in \mathcal{D}_V$ tel que : $(y, j)\ \alpha_P\ (x, i)\}$

Par exemple, si α = indifférent, alors nous obtenons :

$$\begin{aligned} S(c_P[\text{indifférent}, x, i, j], V, D) &= \{y \in \mathcal{D}_V \text{ tel que : } (y, j) \simeq_P (x, i)\} \\ &= \{y \in \mathcal{D}_V \text{ tel que : } (y, j) \simeq_k (x, i), \text{ pour tout } k \in I_{se}\} \\ &= \{y \in \mathcal{D}_V \text{ tel que : } y \in S(c_k[\text{indifférent}, x, i, j], V, D), \text{ pour tout } k \in I_{se}\} \end{aligned}$$

En définitive, $$S(c_P[\text{indifférent}, x, i, j], V, D) = \bigcap_{k \in I_{se}} S(\, c_k[\text{indifférent}, x, i, j], V, D)\ . \qquad (7\text{–}46)$$

En comparant cette équation (7–46) à l'équation (7–40), il suffit de remplacer les concernés I_e et les contraintes souples c_i[indifférent, x] respectivement par les émetteurs I_{se} et les contraintes souples c_k[indifférent, x, i, j]. Il en est de même des autres alternatives. En d'autres termes, $S(c_P[\alpha, x, i, j], V, D)$ obéit aux mêmes règles de construction que $S(c_P[\alpha, x], V, D)$. Cette relation valable pour la règle de Pareto est généralisable à toute règle d'agrégation simple grâce aux relations[182] entre les ensembles $S(c_k[\alpha, x, i, j], V, D)$ et les ensembles $S(c_k[\alpha, (x, i)], D \times I_e)$. Et une théorie équivalente peut donc être construite. D'un point de vue pratique, cette généralisation permet, à partir des algorithmes de consistance d'une règle d'agrégation simple sur un profil simple, d'automatiser directement la conception d'algorithmes de consistance pour cette même règle d'agrégation simple, mais sur un profil type sympathie étendue.

7.4.4.2. Filtrage pour la Règle de Pareto Couplé à l'Axiome d'Identité

La règle de Pareto sur un profil de relations individuelles binaires de préférence étendue ne satisfait pas l'axiome d'identité (§ 6.3.3.2). Lorsque l'axiome d'identité est couplé à la règle de Pareto, nous obtenons la PB-contrainte souple étendue $\{c_{PID}[\alpha, x, i, j]\}_{(\alpha, x, i, j)}$ dont la sémantique est la suivante :

$$S(c_{PID}[\alpha, x, i, j], V, D) = \begin{cases} S_P(\alpha, x, i, j) & \text{si } i \neq j \\ S_i(\alpha, x, i, j) & \text{si } i = j \end{cases}.$$

Un ensemble d'algorithmes de filtrage est alors possible, en fonction de ceux utilisés pour la contrainte souple de Pareto et les contraintes souples associées aux préférences individuelles. Rappelons que cette relation binaire de préférence ne conserve pas la transitivité (voir § 6.3.3.2).

7.4.5. Consistance et Règle d'Équité de Suppes-Sen

La règle d'équité de Suppes-Sen définie au § 6.3.2.1 est une règle d'agrégation étendue générale permettant de modéliser à la fois l'équité et l'efficacité dans un contexte purement ordinal. Nous étudions ici quelques possibilités de résoudre des problèmes de consistance associés.

Considérons donc un ensemble de variables V et un ensemble de domaines D associé à V. La règle d'équité de Suppes-Sen SSE sur l'ensemble des alternatives $\mathcal{D}_V = \times_{v \in V} D(v)$ et des

[182] Voir § 5.2.2.2.

individus concernés I_e s'applique à une relation binaire de préférence étendue $\succcurlyeq$ sur $\mathcal{D}_V \times I_e$, modélisée par une PB-contrainte souple étendue notée $\{c_{\succcurlyeq}[\alpha, x, i, j]\}_{(\alpha, x, i, j)}$. SSE associe à $\succcurlyeq$ des préférences collectives $\succcurlyeq_{SSE}$ formant une relation binaire de préférence sur $\mathcal{D}_V$, représentée par la PB-contrainte souple simple $\{c_{SSE}[\alpha, x]\}_{\alpha, x}$, appelée **contrainte souple de Suppes-Sen** (ou contrainte souple SSE).

Cette sous-section se présente comme suit : les contraintes de Suppes-Sen respectivement en (indifférent, x) et (surclasse, x) sont traitées ensemble à cause de leur sémantique similaire (§ 7.4.5.1). Leurs relations avec certains concepts de la théorie des graphes sont alors présentées et des algorithmes de filtrage sont déduits. Le cas de la préférence stricte est ensuite traité (§ 7.4.5.2) et nous terminons par l'incomparabilité (§ 7.4.5.3).

7.4.5.1. Filtrage Global et Adaptable pour l'Indifférence et le Surclassement

Nous traitons ici les contraintes globales de la contrainte souple de Suppes-Sen ayant la sémantique suivante :

$$S(c_{SSE}[\alpha, x], V, D) = \{y \in \mathcal{D}_V \text{ tel que : } \exists \text{ une permutation } \sigma \text{ de } I_e \text{ avec } (y, i)\ \alpha_{\succcurlyeq}\ (x, \sigma(i))\ \forall\ i \in I_e\}$$

$$S(c_{SSE}[\alpha, x], V, D) = \{y \in \mathcal{D}_V \text{ tel que : } \exists \text{ une permutation } \sigma \text{ de } I_e \text{ avec } y \in S(c_{\succcurlyeq}[\alpha, x, i, \sigma(i)], V, D)\ \forall\ i \in I_e\} \quad (7\text{–}47)$$

Les attitudes α concernées sont l'indifférence, le surclassement et le sousclassement.

Pouvons-nous traduire la permutation en unions et intersections de contraintes globales individuelles ?

Tout d'abord, traitons un exemple simple : Considérons un couple d'individus observés $I_e = \{i, j\}$, le produit cartésien $\mathcal{D}_V$ des domaines D associés à un ensemble de variables V, et une PB-contrainte souple étendue $\{c_{\succcurlyeq}[\alpha, x, i, j]\}_{(\alpha, x, i, j)}$ modélisant une relation binaire de préférence étendue $\succcurlyeq$ sur $\mathcal{D}_V \times I_e$. La figure ci-dessous illustre graphiquement les différentes intersections possibles entre les ensembles de solutions réalisables des contraintes globales de $\{c_{\succcurlyeq}[\alpha, x, i, j]\}_{(\alpha, x, i, j)}$ lorsque l'attitude α et la solution x sont fixées :

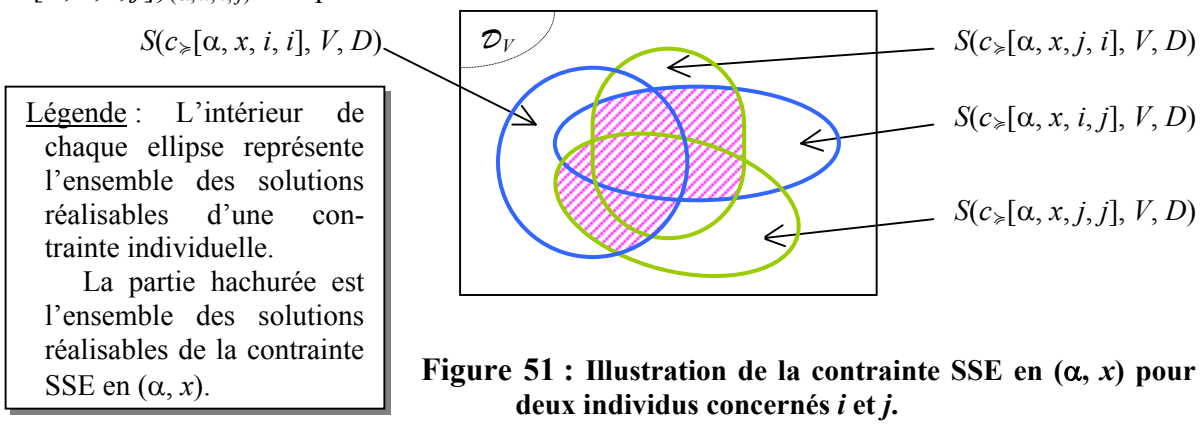


Figure 51 : Illustration de la contrainte SSE en (α, x) pour deux individus concernés *i* et *j*.

La formule (7–47) indique que pour qu'une solution y soit réalisable pour la contrainte $c_{SSE}[\alpha, x]$, elle doit être située :

(a) dans l'intersection de $S(c_{\succcurlyeq}[\alpha, x, i, i], V, D)$ et $S(c_{\succcurlyeq}[\alpha, x, j, j], V, D)$, c'est-à-dire que x et y sont liés par la permutation identité $\begin{pmatrix} i & j \\ i & j \end{pmatrix}$, ou

(b) dans l'intersection de $S(c_{\succcurlyeq}[\alpha, x, i, i], V, D)$ et $S(c_{\succcurlyeq}[\alpha, x, i, i], V, D)$, c'est-à-dire que x et y sont liés par la transposition $\begin{pmatrix} i & j \\ j & i \end{pmatrix}$.

❖ Couverture des Sommets de Graphes Bipartis pour un Filtrage Adaptable

Voyons comment l'ensemble $S(c_{\mathrm{SSE}}[\alpha, x], V, D)$ peut être approximé en utilisant des opérations ensemblistes entre les contraintes globales de la contrainte souple étendue individuelle $\{c_{\succcurlyeq}[\alpha, z, i, j]\}_{(\alpha, z, i, j)}$. Nous avons la proposition suivante :

Proposition 25 : Considérons une PB-contrainte souple étendue $\{c_{\succcurlyeq}[\alpha, z, i, j]\}_{(\alpha, z, i, j)}$ représentant une relation binaire de préférence étendue $\succcurlyeq$ d'un individu arbitraire sur le produit cartésien des domaines D associées à un ensemble de variables V, et sur un ensemble d'individus observés I_e ($\mathrm{card}(I_e) \geq 2$). Alors la contrainte de Suppes-Sen associée à l'attitude $\alpha \in \{\text{indifférent, surclasse, sousclasse}\}$ et à la solution $x \in \mathcal{D}_V$ vérifie l'inclusion suivante :

$$S(c_{\mathrm{SSE}}[\alpha, x], V, D) \subseteq \left(\bigcap_{i \in I_e} \bigcup_{j \in I_e} S(c_{\succcurlyeq}[\alpha, x, i, j], V, D) \right) \cap \left(\bigcap_{j \in I_e} \bigcup_{i \in I_e} S(c_{\succcurlyeq}[\alpha, x, i, j], V, D) \right) \quad (7\text{–}48)$$

Si $\mathrm{card}(I_e) = 2$ alors l'inclusion inverse est vérifiée.

Avant de donner une preuve, décortiquons la formule (7–48). Le second membre est formé de deux termes, inversant les rôles de i et de j. Le premier terme recherche pour chaque i au moins un j pour lequel $S(c_{\succcurlyeq}[\alpha, x, i, j], V, D)$ est non vide. Le second terme recherche pour chaque j au moins un i pour lequel $S(c_{\succcurlyeq}[\alpha, x, i, j], V, D)$ est non vide.
Sur la Figure 51, la formule (7–48) indique qu'une solution y doit être située dans l'une des 7 intersections de contraintes individuelles formant la partie hachurée. Ce schéma démontre graphiquement l'égalité, lorsque $\mathrm{card}(I_e) = 2$, entre le domaine réalisable de la contrainte $c_{\mathrm{SSE}}[\alpha, x]$ et le second membre de la formule (7–48). Donnons maintenant une première preuve de la Proposition 25 :

Preuve : Cette preuve est assez directe. Tout d'abord, une permutation σ est une bijection. Ce qui est équivalent à dire que : (a) pour tout $i \in I_e$, il existe un $j = \sigma(i) \in I_e$ tel que $y \in S(c_{\succcurlyeq}[\alpha, x, i, j], V, D)$; et (b) pour tout $l \in I_e$, il existe un $k = \sigma^{-1}(l) \in I_e$ vérifiant $y \in S(c_{\succcurlyeq}[\alpha, x, k, l], V, D)$. Par conséquent en supprimant le lien $j = \sigma(i)$ entre i et j, et le lien $k = \sigma^{-1}(l)$ entre k et l, toute référence à la permutation σ peut être supprimée, et nous obtenons une condition nécessaire :

$$S(c_{\mathrm{SSE}}[\alpha, x], V, D) \subseteq \left\{ y \in \mathcal{D}_V \text{ tel que :} \left| \begin{array}{l} \forall\, i \in I_e, y \in \bigcup_{j \in I_e} S(c_{\succcurlyeq}[\alpha, x, i, j], V, D), \text{ et} \\ \forall\, l \in I_e, y \in \bigcup_{k \in I_e} S(c_{\succcurlyeq}[\alpha, x, k, l], V, D) \end{array} \right. \right\}$$

Ce qui est équivalent au second membre de la formule (7–48).

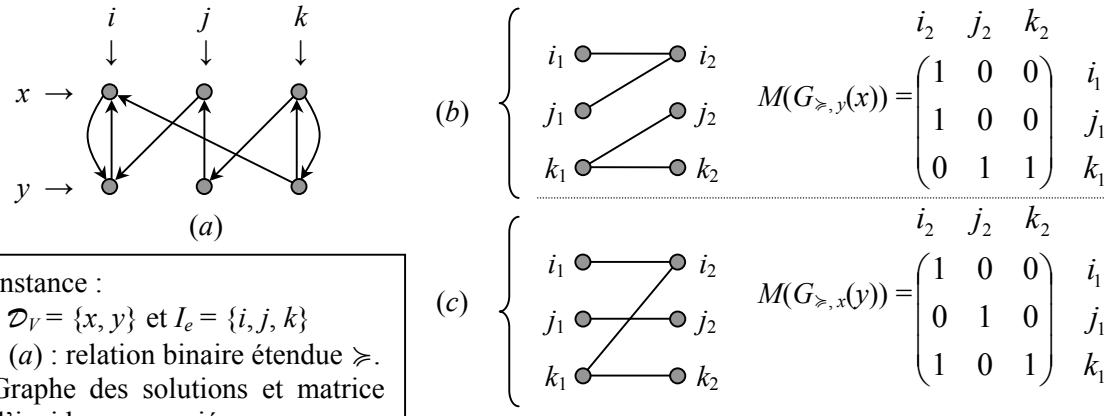
Lorsque $\mathrm{card}(I_e) = 2$, alors l'existence d'un ensemble non vide pour chaque i et l implique qu'il existe une permutation de I_e vérifiant la formule (7–47). En d'autres termes, toute intersection d'ensembles satisfaisant (7–48) induit l'existence de la permutation identité ou la transposition. Cette implication est directe. Nous la laissons en exercice au lecteur. ❑

La formule (7–48) permet de construire un premier algorithme de filtrage, en utilisant le second membre pour définir un système de contraintes : Chaque union du premier terme est équivalente à un opérateur de cardinalité $\#_i(v, [1, \ldots, \mathrm{card}(I_e)], \{V, D, C_i\})$ sur un ensemble de contraintes C_i formé de $\mathrm{card}(I_e)$ contraintes individuelles de la relation binaire de préférence étendue $\succcurlyeq$: $C_i = \{c_{\succcurlyeq}[\alpha, x, i, j], \forall\, i \in I_e\}$. De la même manière, un opérateur de cardinalité est associé à chaque j, pour le second terme de la formule (7–48). Nous obtenons alors un système de contraintes $\mathcal{R}_{\#}$ formé de $2 \cdot \mathrm{card}(I_e)$ opérateurs de cardinalité. Par conséquent tout algorithme de filtrage sur $\mathcal{R}_{\#}$ supprime des valeurs et/ou combinaisons de valeurs inconsistantes avec la contrainte SSE en (α, x).

Remarquons que chaque contrainte individuelle $c_{\succcurlyeq}[\alpha, x, i, j]$ occurre dans exactement deux opérateurs de cardinalité différents : $\#_i$ et $\#_j$. Cette particularité pourra être exploitée pour optimiser les algorithmes de filtrage basés sur $\mathcal{K}_{\#}$ pour $c_{\mathrm{SSE}}[\alpha, x]$.

Pour mieux étudier ce type de contraintes, nous allons nous ramener à un problème de graphe de la manière suivante :

Considérons un couple $(\alpha, x) \in \mathcal{PR}(AF(RBP)) \times \mathcal{D}_V$. À toute solution $y \in \mathcal{D}_V$ associons un graphe biparti $G_{\alpha, x}(y) = (U_1 \cup U_2, A)$ obtenu en définissant pour chaque individu i un sommet $u_1(i)$ de U_1 et un sommet $u_2(i)$ de U_2 ; et un couple de sommets $(u_1(i), u_2(j))$ étant une arête si et seulement si $y \in S(c_{\succcurlyeq}[\alpha, x, i, j], V, D)$. Le graphe $G_{\alpha, x}(y)$ est appelé **graphe de la solution** y en (α, x). La Figure 52 montre l'exemple d'une relation binaire de préférence étendue sur 3 individus observés et sur un ensemble de 2 solutions. Les graphes de la solution x en (surclasse, y) et de la solution y en (surclasse, x) sont donnés, ainsi que leur matrice d'incidence[183].



Instance :
$\mathcal{D}_V = \{x, y\}$ et $I_e = \{i, j, k\}$
(a) : relation binaire étendue $\succcurlyeq$.
Graphe des solutions et matrice d'incidence associée :
(b) : $G_{\succcurlyeq, y}(x)$ et $M(G_{\succcurlyeq, y}(x))$
(c) : $G_{\succcurlyeq, x}(y)$ et $M(G_{\succcurlyeq, x}(y))$

Figure 52 : Illustration des PB-contraintes souples étendues dont les solutions sont caractérisées par une permutation.

Notons $n = \mathrm{card}(I_e) = \mathrm{card}(U_1) = \mathrm{card}(U_2)$. Un graphe biparti G sur deux groupes de n sommets est dit **valide** pour la contrainte SSE en (α, x) si et seulement s'il existe une solution $y \in \mathcal{D}_V$ tel que G soit le graphe de y en (α, x) <u>et</u> que $y \in S(c_{\mathrm{SSE}}[\alpha, x], V, D)$. Notons $BG(SSE, \alpha, x, n)$ l'ensemble des graphes bipartis valides pour la contrainte SSE en (α, x). Plus généralement, à tout sous-ensemble $\mathcal{A}$ de $\mathcal{D}_V$ peut être associé l'ensemble $BG(\mathcal{A}, \alpha, x, n)$ des graphes bipartis valides pour $\mathcal{A}$ en (α, x) :

$$G \in BG(\mathcal{A}, \alpha, x, n) \quad \Leftrightarrow \quad \exists\, y \in \mathcal{D}_V \text{ tel que : } y \in \mathcal{A} \text{ et } G_{\alpha, x}(y) = G.$$

Un graphe non orienté simple $G = (U(G), A(G))$, avec $\mathrm{card}(U(G)) = m$, peut être représenté par un **vecteur d'incidence** $x_G \in \{0, 1\}^{m^2}$ défini par :

$$x_G(e) = \begin{cases} 1 & \text{si } e \in A(G) \\ 0 & \text{sinon.} \end{cases}$$

m^2 est le nombre d'arêtes potentielles pour être dans G. Dans le cas des graphes bipartis sur deux groupes respectivement de $n1$ et $n2$ sommets, le vecteur d'incidence x_G peut être défini sur $\{0, 1\}^{n1 \cdot n2}$ au lieu de $\{0, 1\}^{(n1+n2)^2}$ car les arêtes candidates sont uniquement celles où chaque extrémité est dans un groupe de sommets différent[184].

[183] La **matrice d'incidence** $M(G)$ d'un graphe biparti $G = (U_1 \cup U_2, A)$ est une matrice à coefficients 0 ou 1, où chaque ligne i correspond à un sommet i de U_1, chaque colonne j à un sommet j de U_2 ; et le coefficient $M(G)_{ij}$ de la matrice est égal à 1 ssi (i, j) est une arête de G.

[184] Remarquons que la matrice d'incidence d'un graphe biparti est équivalente au vecteur d'incidence de ce graphe : Lorsque $G = (U_1 \cup U_2, A)$ est un graphe biparti, alors pour tout $e = (i, j) \in U_1 \times U_2$, $x_G(e) = M(G)_{ij}$.

Ainsi, à tout sous-ensemble $\mathcal{A}$ de $\mathcal{D}_V$ est associé un ensemble de vecteurs de $\{0, 1\}^{n^2}$, avec n = card(I_e). L'enveloppe convexe (§ 2.1.3.1) des vecteurs d'incidence des graphes valides de $\mathcal{A}$ forme un polytope $P_{\mathcal{A}}(n)$ de $^{n^2}$.
La proposition suivante montre que pour certaines descriptions par des inégalités du polytope $P_{\mathcal{A}}(n)$, il existe une description équivalente de $\mathcal{A}$ par des unions et des intersections d'ensembles $S(c_{\succcurlyeq}[\alpha, x, i, j], V, D)$ pour tout i, j de I_e.

Proposition 26 : Considérons un septuplet (V, D, I_e, $\mathcal{A}$, $\succcurlyeq$, α, x) formé d'un ensemble de variables V, un ensemble de domaines D associé à V, un ensemble d'individus observés I_e, un sous-ensemble $\mathcal{A}$ du produit cartésien des domaines $\mathcal{D}_V$, une relation binaire de préférence étendue $\succcurlyeq$ sur $\mathcal{D}_V \times I_e$, une attitude α de $\mathcal{PR}(AF(RBP))$ et un élément x de $\mathcal{D}_V$. Notons n = card(I_e).
Soit $A^{\mathcal{A}}y \geq b^{\mathcal{A}}$ un système d'inégalités linéaires telle que $A^{\mathcal{A}}$ est une matrice $m \times n^2$ à coefficients 0 ou 1, y un vecteur de dimension n^2 et b un vecteur entier de dimension m. En notant a_{kl} le coefficient de la k^e ligne et de la l^e colonne de la matrice $A^{\mathcal{A}}$, et $b^{\mathcal{A}}(k)$ le k^e coefficient du vecteur $b^{\mathcal{A}}$, Alors les deux assertions suivantes sont équivalentes :
(a) $A^{\mathcal{A}}y \geq b^{\mathcal{A}}$ est une description par des inégalités du polytope $P_{\mathcal{A}}(n)$
(b) $\mathcal{A} = \bigcap_{k=1}^{m} \bigcup_{\substack{B \subseteq \{l \mid a_{kl}=1\} \cap (U_1 \times U_2) \\ \text{et card}(B)=b^{\mathcal{A}}(k)}} \bigcap_{e \in B} S(c_{\succcurlyeq}[\alpha, x, i, j], V, D)$.
avec (i, j) le couple d'individus (c'est-à-dire de sommets) associé à l'arête e :
$$e = (u_1(i), u_2(j)) \in U_1 \times U_2 \text{ et } (i, j) \in I_e^2.$$

Preuve : La démonstration est donnée en Annexe B3.

Remarque : Cette proposition ne nécessite même pas que la description par des inégalités du polytope $P_{\mathcal{A}}(n)$ corresponde aux facettes de ce polytope. Juste une description délimitant clairement les vecteurs d'incidence à coefficients en 0 et 1, et le reste des vecteurs entiers de $^{n^2}$.

Cette proposition nous donne des arguments de poids pour rechercher une description du polytope $P_{\mathcal{A}}(n)$ par des inégalités : cette description traduit directement $\mathcal{A}$ par des unions et intersections d'ensembles $S(c_{\succcurlyeq}[\alpha, x, i, j], V, D)$, pour tout i, j de I_e. Ainsi, dans le reste de cette section, nous allons caractériser les graphes bipartis valides pour l'ensemble des solutions $S_{EC}(V, D, I_e, \alpha, x)$ définies par le second membre de la formule (7–48) et pour la contrainte SSE en (α, x). Notons $P_\Gamma(n)$ et $P_{\text{SSE}}(n)$ les polytopes associés aux ensembles respectivement $S_{EC}(V, D, I_e, \alpha, x)$ et $S(c_{\text{SSE}}[\alpha, x], V, D)$.
D'abord rappelons quelques définitions de la théorie des graphes (SAKAROVITCH [1984*a*]) :

Un sous-ensemble $\hat{A}$ des arêtes d'un graphe sans boucle $G = (U, A)$ est un **recouvrement des sommets** de G **par les arêtes** (ou couverture des sommets de G, *edge cover*) si tout sommet u est adjacent à une arête de $\hat{A}$ au moins.
La description des facettes du polytope $P_{EC}(G)$ des recouvrements des sommets de G par les arêtes est connue pour tout graphe[185]. Dans le cas des graphes bipartis le nombre de facettes est polynomial en le nombre de sommets du graphe, et est décrit par le système d'inégalités linéaires suivant :

[185] Voir EDMONDS J. & JOHNSON E. [1970] – Matching, a Well Solved Class of Integer Linear Programs – *in Combinatorial Structures and Their Applications*, Gordon and Breach, New-York : 89-92. Voir aussi CARR *et al.* [2000] ou encore SCHRIJVER [2003].

$$P_{EC}(G) = \left\{ x \in \quad^{\text{card}(A)} \text{ tel que : } \left| \begin{array}{l} \sum_{e \in \delta(u)} x(e) \geq 1 \text{ pour tout } u \in U_1 \cup U_2 \\ 0 \leq x(e) \leq 1 \text{ pour tout } e \in A \end{array} \right. \right\}$$

avec $G = (U_1 \cup U_2, A)$ et

$\delta(u)$ l'ensemble des arêtes adjacentes au sommet u ($\Leftrightarrow \delta(u) = \{e \in A$ tel que : $u \in e\}$).

Notons $K_{n1,n2}$ le graphe biparti complet sur deux groupes de sommets de cardinaux respectifs $n1$ et $n2$ sommets ($\Leftrightarrow K_{n1,n2} = (U_1 \cup U_2, U_1 \times U_2)$, avec $\text{card}(U_1) = n1$ et $\text{card}(U_2) = n2$), et $P_{BG(EC)}(n1, n2)$ le polytope des graphes bipartis sur deux groupes de sommets de cardinalités respectives $n1$ et $n2$.

Le polytope $P_{EC}(G)$ permet de répondre à la question : Quels sont les recouvrements des sommets par les arêtes d'un graphe biparti ? Mais la question qui nous intéresse est : Quels sont les graphes bipartis sur le couple d'ensembles de sommets (U_1, U_2) admettant au moins un recouvrement des sommets par les arêtes ? Le lemme suivant permet d'y répondre :

Lemme 1 : L'ensemble des graphes bipartis, admettant au moins un recouvrement des sommets par les arêtes, est en bijection avec l'ensemble des recouvrements des sommets du graphe biparti complet par ses arêtes. En d'autres termes : $P_{BG(EC)}(n1, n2) = P_{EC}(K_{n1,n2})$.

Preuve : La démonstration est immédiate : Il suffit de constater que toute couverture des sommets de $K_{n1,n2}$ induit un graphe partiel, lequel est un graphe contenant un recouvrement des sommets qui est son propre ensemble d'arêtes ; et de remarquer que tout graphe admettant une couverture de ses sommets est un graphe partiel du graphe biparti complet. ❑

Ce lemme permet alors d'aboutir à la proposition suivante reliant les solutions réalisables pour $S_{EC}(V, D, I_e, \alpha, x)$ aux recouvrements des sommets d'un graphe biparti par ses arêtes :

Proposition 27 : $S_{EC}(V, D, I_e, \alpha, x) = \{y \in \mathcal{D}_V$ tel que : $x_{G_{\alpha,x}(y)} \in P_{EC}(K_{n,n})\}$.

Preuve : Il suffit d'appliquer la Proposition 26, et nous arrivons au résultat. ❑

Ainsi, connaître le second membre de la formule (7–48) de la Proposition 25 c'est connaître les différents recouvrements des sommets par les arêtes du graphe biparti complet sur deux groupes de n sommets.

Remarque 5 : Les combinaisons (par des unions et intersections) d'ensembles $S(c_{\geqslant}[\alpha, x, i, j], V, D)$ qui sont susceptibles de contenir des solutions réalisables pour $\mathcal{A}$, sont <u>indépendantes</u> de l'état réel des ensembles $S(c_{\geqslant}[\alpha, x, i, j], V, D)$. Par exemple, pour une instance donnée $(V, D, I_e, \geqslant, \alpha, x)$, certains ensembles $S(c_{\geqslant}[\alpha, x, i, j], V, D)$ peuvent être vides. Et donc tous les graphes décrits par $P_{BG(EC)}(n, n)$ ne sont pas forcément valides pour $S_{EC}(V, D, I_e, \alpha, x)$. Plus précisément, les graphes valides pour $S_{EC}(V, D, I_e, \alpha, x)$, sont contenus dans l'ensemble des graphes décrits par $P_{BG(EC)}(n, n)$:

$BG(S_{EC}(V, D, I_e, \alpha, x), \alpha, x, n) \subseteq \{G$ tel que : $x_G \in P_{BG(EC)}(n, n)\} \Leftrightarrow P_\Gamma(n) \subseteq P_{BG(EC)}(n, n)$

Par contre, ce qui fait la force de la Proposition 27, c'est qu'il y a égalité entre l'ensemble $S_{EC}(V, D, I_e, \alpha, x)$ et l'ensemble $\{y \in \mathcal{D}_V$ tel que : $x_{G_{\alpha,x}(y)} \in P_{BG(EC)}(n, n)\}$, et ce quelque soit l'instance $(V, D, I_e, \geqslant, \alpha, x)$. Et finalement, c'est ce qui est important.

Revenons maintenant à l'inclusion (7–48). Elle traduit simplement que les graphes valides pour la contrainte SSE en (α, x) doivent admettre un recouvrement des sommets par les arêtes, mais pas uniquement cela. Étudions cela de plus près :

❖ Graphes Bipartis Contenant un Couplage Parfait pour un Filtrage Global Adaptable

Le théorème de BIRKHOFF-VON NEUMANN[186] nous donne une première réponse. Ce théorème, énoncé de manière informelle au § 6.2.3.2, caractérise le polytope d'affectation B_n, c'est-à-dire le polytope des vecteurs d'incidence associés à l'ensemble des couplages parfaits du graphe biparti complet, comme étant le polytope de toutes les matrices doublement stochastiques. En d'autres termes, il existe une bijection de l'ensemble des couplages parfaits du graphe biparti complet sur deux groupes de n sommets vers l'ensemble des permutations de n éléments. Par conséquent, y est une solution réalisable de la contrainte SSE en (α, x) si et seulement si le graphe de cette solution en (α, x) contient un couplage parfait :

$$S(c_{\mathrm{SSE}}[\alpha, x], V, D) = \{y \in \mathcal{D}_V \text{ tel que : } P_{PM}(G_{\alpha, x}(y)) \neq \varnothing\} \qquad (7\text{–}49)$$

avec $P_{PM}(G)$ le polytope des couplages parfaits du graphe biparti G, dont la description des facettes est (voir par exemple CARR *et al.* [2000], SCHRIJVER [2003]) :

$$P_{PM}(G) = \left\{ x \in \quad^{\mathrm{card}(A)} \text{ tel que : } \left| \begin{array}{l} \sum_{e \in \delta(u)} x(e) = 1 \text{ pour tout } u \in U_1 \cup U_2 \\ 0 \leq x(e) \leq 1 \text{ pour tout } e \in A \end{array} \right. \right\},$$

pour $G = (U_1 \cup U_2, A)$ et

$\delta(u)$ l'ensemble des arêtes adjacentes au sommet u ($\Leftrightarrow \delta(u) = \{e \in A$ tel que : $u \in e\}$).

Écrit autrement, avec le polytope $P_{BG(PM)}(n)$ des graphes bipartis sur deux groupes de n sommets contenant un couplage parfait :

$$S(c_{\mathrm{SSE}}[\alpha, x], V, D) = \{y \in \mathcal{D}_V \text{ tel que : } x_{G_{\alpha,x}(y)} \in P_{BG(PM)}(n)\} \qquad (7\text{–}50)$$

Comparé à la description du polytope $P_{EC}(G)$ des recouvrements des sommets d'un graphe biparti G par des arêtes, le polytope $P_{PM}(G)$ des couplages parfaits de G utilise l'opérateur d'égalité au lieu de l'opérateur supérieur ou égal, pour l'inégalité. Ainsi tout graphe biparti admettant un couplage parfait admet une couverture de ses sommets : $P_{PM}(G) \subseteq P_{EC}(G)$ et $S(c_{\mathrm{SSE}}[\alpha, x], V, D) \subseteq S_{EC}(V, D, I_e, \alpha, x)$. Ce qui est une autre démonstration de la Proposition 25. Mais, un graphe biparti admettant une couverture des sommets ne contient pas toujours un couplage parfait ! ! L'exemple de la Figure 52 illustre bien ce cas : $S(c_{\mathrm{SSE}}[\mathsf{surclasse}, x], V, D) = \{y\}$ alors que $S_{EC}(V, D, I_e, \mathsf{surclasse}, x) = \{x, y\}$. Si on utilise les matrices d'incidence, cela veut dire qu'il ne suffit pas que la matrice d'incidence des solutions ait la somme des coefficients de chaque ligne et de chaque colonne supérieure ou égale à 1. L'équivalence n'est obtenue que dans le cas où $\mathsf{card}(I_e) = 2$. L'équivalence peut être obtenue pour n'importe quel nombre d'individus sur I_e, si des restrictions sont effectuées sur $\mathcal{D}_V$ ou sur la relation binaire de préférence étendue $\succcurlyeq$. On entre alors dans le cadre d'une analyse axiomatique (§ 4.2.2). Nous n'irons pas plus loin dans ce sens.

Y a-t-il une relation entre l'ensemble des couplages dans le graphe biparti complet et l'ensemble des graphes bipartis contenant un couplage parfait ? : $B_n \subseteq P_{BG(PM)}(n)$, en rappelant que B_n et $P_{PM}(K_{n,n})$ désignent le même polytope. De plus, l'ensemble des couplages parfaits dans le graphe biparti complet a une intersection non vide avec l'ensemble des graphes valides pour la contrainte SSE en (α, x) : $P_{\mathrm{SSE}}(n) \cap B_n \neq \varnothing$ si $S(c_{\mathrm{SSE}}[\alpha, x], V, D) \neq \varnothing$ et s'il existe au moins un graphe dont l'ensemble des arêtes forment un couplage parfait (il n'y a pas d'arêtes supplémentaires en dehors de celles du couplage). Mais $P_{\mathrm{SSE}}(n)$ n'est pas inclus dans B_n.

Les couplages ont été l'un des concepts en théorie des graphes les plus étudiés. Le fameux théorème de Koënig-Hall, dont l'énoncé est donné ci-dessous, en est un exemple. Il nous

[186] Voir par exemple : BIRKHOFF G. [1946] – Très Observaciones sobre el Algebra Lineal – *Univ. Nac. Tucuman Rev. Ser. A* 5 : 147-151.

permettra de formuler la contrainte SSE en (α, x) directement en fonction des contraintes individuelles :

Théorème 6 (de Koënig-Hall) : Il existe dans le graphe biparti $G = (U_1 \cup U_2, A)$ un couplage saturant tous les sommets de U_1 si et seulement si on a :

$$\mathsf{card}(\{u_2 \in U_2 \text{ tel que : } (u_1, u_2) \in A \text{ et } u_1 \in B\}) \geq \mathsf{card}(B) \qquad \text{pour tout } B \subseteq U_1.$$

En d'autres termes, les voisins de tout sous-ensemble de sommets B de U_1 ont un nombre de sommets supérieur à celui de B. Ce théorème ne permet pas, à partir d'un graphe biparti de construire un couplage parfait. Mais il permet de répondre entre autre aux questions : Quels sont les graphes bipartis saturant un ensemble de sommets donné, Quels sont les graphes bipartis contenant un couplage parfait, etc..
Le théorème de Koënig-Hall donne une traduction directe de $S(c_{\mathrm{SSE}}[\alpha, x], V, D)$ en unions et intersections d'ensembles $S(c_{\succcurlyeq}[\alpha, x, i, j], V, D)$ pour tout $(i, j) \in I_e^2$:

Corollaire : $$S(c_{\mathrm{SSE}}[\alpha, x], V, D) = \bigcap_{\varnothing \neq J_1 \subseteq I_e} \bigcup_{\substack{J_2 \subseteq I_e \text{ tel que} \\ \mathsf{card}(J_2) = \mathsf{card}(J_1)}} \bigcap_{j \in J_2} \bigcup_{i \in J_1} S(c_{\succcurlyeq}[\alpha, x, i, j], V, D).$$

Preuve : Voir Annexe B3.

Cette égalité peut servir de base à des filtrages. L'approche directe effectuée pour la couverture de sommets de $K_{n,n}$ (voir Proposition 25 et conséquences), aboutit à une complexité largement exponentielle en la taille des données (V, D, I_e, $\succcurlyeq$, α, x) : Cette approche directe consiste à transformer les unions en opérateurs de cardinalité et les intersections en systèmes de contraintes[187]. Les relations entre ensembles $S(c_{\succcurlyeq}[\alpha, x, i, j], V, D)$ peuvent être méthodiquement utilisées pour éviter les redondances et réduire la complexité. Nous n'irons pas plus loin dans ce sens. Nous avons décidé de prospecter une autre voie pour l'identification pratique de $S(c_{\mathrm{SSE}}[\alpha, x], V, D)$.

❖ Énumération des Couplages Parfaits pour un Filtrage Global Adaptable

Il existe d'autres manières d'identifier les solutions réalisables de la contrainte SSE en (α, x) $\in$ {indifférent, surclasse, sousclasse} $\times\ \mathcal{D}_V$. Par exemple, en cherchant une description par des inégalités linéaires du polytope $P_{BG(PM)}(n)$ des graphes bipartis admettant un couplage parfait. Nous avons décidé de tirer partie de la formule (7–49). Nous pouvons passer par l'énumération de l'ensemble des couplages parfaits des graphes valides de $S(c_{\mathrm{SSE}}[\alpha, x], V, D)$. Comment ? Tout d'abord, un couplage parfait est un sous-ensemble de sommets. Cette propriété peut être exploitée pour exprimer la contrainte collective en fonction des contraintes individuelles.

Soit M un couplage parfait de $K_{n,n}$, et notons $BG_{\mathrm{SSE}}(M)$ l'ensemble des graphes valides de $S(c_{\mathrm{SSE}}[\alpha, x], V, D)$ tel que M soit l'un de leurs couplages parfaits :

---

[187] Ainsi,

(a) l'union la plus interne est remplacée par un opérateur de cardinalité $\#_{(J1,j)}(v, [1, \ldots, \mathsf{card}(J_1)], \{V, D, C_{(J1,j)}\})$ sur un ensemble de contraintes $C_{(J1,j)}$ formé de $\mathsf{card}(J_1)$ contraintes individuelles de la relation binaire de préférence étendue $\succcurlyeq$: $C_{(J1,j)} = \{c_{\succcurlyeq}[\alpha, x, i, j], \forall\, i \in J_1\}$;

(b) l'intersection la plus interne est remplacée par l'opérateur de cardinalité $\#_{(J1,J2)}(v, \{\mathsf{card}(J_2)\}, \{V, D, C_{(J1,J2)}\})$ sur l'ensemble de contraintes $C_{(J1,J2)}$ formé des $\mathsf{card}(J_2)$ opérateurs de cardinalité $C_{(J1,j)} = \{\#_{(J1,j)}, \forall\, j \in J_2\}$;

(c) l'union la plus externe est remplacée par un opérateur de cardinalité $\#_{J1}(v, [1, \ldots, m(n, J_1)], \{V, D, C_{J1}\})$ sur un ensemble de contraintes C_{J1} formé des $m(n, J_1)$ opérateurs de cardinalité $C_{J1} = \{\#_{(J1,J2)}, \forall\, J_2 \in I_e$ tel que $\mathsf{card}(J_2) = \mathsf{card}(J_1)\}$, avec $m(n, J_1) = \dfrac{n!}{(n - \mathsf{card}(J_1))!\mathsf{card}(J_1)!}$; et

(d) l'intersection la plus externe est remplacée par un système de contraintes formé de $2^n - 1$ opérateurs de cardinalité $\#_{J1}$.

$$BG_{\mathrm{SSE}}(M) = \{G \in BG(SSE, \alpha, x, n) \text{ tel que : } M \subseteq A(G)\}$$

Alors nous avons la relation suivante :

$$\bigcup_{G \in BG_{\mathrm{SSE}}(M)} \bigcap_{\substack{(i,j) \in I_e^2 \text{ tel que :} \\ (u_1(i), u_2(j)) \in A(G)}} S(c_{\geqslant}[\alpha, x, i, j], V, D) = \bigcap_{\substack{(i,j) \in I_e^2 \text{ tel que :} \\ (u_1(i), u_2(j)) \in M}} S(c_{\geqslant}[\alpha, x, i, j], V, D).$$

En d'autres termes, les solutions réalisables de $S(c_{\mathrm{SSE}}[\alpha, x], V, D)$, dont les graphes bipartis valides associés contiennent un couplage parfait donné, sont égales aux solutions de $\mathcal{D}_V$ satisfaisant simultanément les contraintes individuelles identifiées par les arêtes du couplage.

En conséquence, notons donc $PM(SSE, \alpha, x, n)$ les différents couplages parfaits tels qu'il existe pour chacun au moins un graphe valide l'admettant comme sous-ensemble d'arêtes :

$$PM(SSE, \alpha, x, n) = \left\{ M \subseteq U_1 \times U_2 \text{ tel que : } \left| \begin{array}{l} x_M \in B_n \\ \exists\, G \in BG(SSE, \alpha, x, n) \text{ avec } M \subseteq A(G) \end{array} \right. \right\}.$$

Ces **couplages parfaits** sont qualifiés de **valides** en (α, x) pour SSE. Alors nous avons l'égalité suivante :

Propriété 8 : $S(c_{\mathrm{SSE}}[\alpha, x], V, D) = \bigcup_{M \in PM(SSE, \alpha, x, n)} \bigcap_{\substack{(i,j) \in I_e^2 \text{ tel que :} \\ (u_1(i), u_2(j)) \in M}} S(c_{\geqslant}[\alpha, x, i, j], V, D).$

La preuve est immédiate.

Cette propriété peut être utilisée pour établir des algorithmes de filtrage. Tout d'abord, rappelons que le nombre de couplages parfaits sur le graphe biparti $K_{n,n}$ est $n!$, c'est-à-dire le nombre de sommets du polytope d'affectation B_n d'ordre n (§ 6.2.3.2). L'égalité ci-dessus est basée sur une liste des couplages parfaits valides pour $S(c_{\mathrm{SSE}}[\alpha, x], V, D)$. Cette liste n'étant pas explicitement connue, il est nécessaire d'utiliser des algorithmes d'énumération de couplages parfaits. Une énumération méthodique des couplages parfaits couplée intelligemment avec la suppression de valeurs de domaines peut réduire significativement la complexité pratique de l'établissement d'un niveau de consistance. Comment procéder ? L'essentiel des algorithmes de listing de couplages parfaits d'un graphe sont basés sur une recherche arborescente (backtracking). Le même type d'algorithmes peut être utilisé pour notre problème.

Par exemple, notre recherche peut être basée sur une arborescence à n niveaux, possédant $n!$ feuilles. Chaque niveau de l'arborescence est associé à un sommet de U_1. À un niveau donné, chaque nœud est un sommet de U_2 auquel le sommet de U_1 associé au niveau peut être couplé. La Figure 53 donne un exemple d'une telle arborescence pour 3 individus. Chaque chemin de la racine à une feuille désigne un couplage parfait. Ainsi, la branche indiquée en gras (racine, j_2, k_2, j_2) représente le couplage parfait $\{(i_1, j_2), (j_1, k_2), (k_1, i_2)\}$.

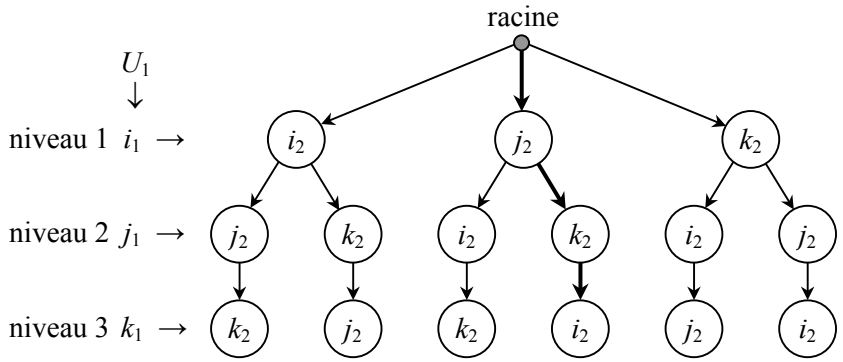


Figure 53 : Arborescence de recherche des couplages parfaits pour trois individus observés $I_e = \{i, j, k\}$.

Un algorithme de filtrage se réduit à parcourir cette arborescence de façon à éviter les redondances de calcul. Si on utilise la stratégie de parcours en profondeur d'abord (DFS :

depth-first search), nous obtenons l'Algorithme 10 de filtrage. D'autres stratégies de recherche autre que DFS peuvent être utilisées comme celles basées sur le retour arrière intelligent. Elles permettront de réduire de manière plus drastique, les redondances de vérification de consistance. Par exemple, si certaines combinaisons de contraintes individuelles aboutissent obligatoirement à une inconsistance (explication d'incohérence § 3.2.3.5), il n'est pas nécessaire de les vérifier à nouveau.
Comme pour les contraintes de Pareto (Algorithme 9 par exemple), selon la méthode filtrer($\mathcal{R}_0$) utilisée, nous obtenons différents niveaux de consistance (algorithme adaptable § 7.3.1). Et si filtrer correspond à un algorithme établissant la consistance inverse globale (GIC) alors l'Algorithme 10 retourne un ensemble de domaines GIC.

❖ Conclusion et Perspectives partielles

Quatre approches de la formulation de contraintes SSE en fonction des contraintes individuelles ont été survolées ou approfondies ici. Une première basée sur une approximation par les recouvrements de sommets, une seconde basée sur les facettes du polytope des graphes bipartis admettant un couplage parfait, une troisième utilisant la caractérisation de Koënig-Hall, et enfin une quatrième basée sur l'énumération des couplages parfaits. Pour le moment, la question d'une formulation polynomiale, comme pour les graphes contenant une couverture des sommets, de la contrainte SSE en fonction des contraintes individuelles reste en suspens.

Discutons maintenant des perspectives :

La Figure 52 nous a présenté le cas d'une relation binaire de préférence étendue n'ayant aucune propriété structurale apparente. C'est aussi dans ce cadre général qu'a été étudiée la contrainte SSE en (α, x) avec $\alpha \in$ {indifférent, surclasse, sousclasse}. Mais, si cette relation binaire de préférence étendue est transitive ou inter-solution transitive (§ 6.3.2.1) les propriétés induites sur les contraintes globales associées peuvent être avantageusement exploitées pour réduire significativement le temps de résolution. En effet, seuls certains graphes bipartis peuvent être obtenus à partir de relations binaires de préférences étendues possédant de telles propriétés structurelles.

Lorsque le problème de consistance s'insère dans une hiérarchie agrégative (§ 4.2.1.5) ou/et dans un processus décisionnel itératif plus général, l'utilisation de l'apprentissage (§ 3.2.2.6) permet de réduire le temps de calcul global.

Les performances de résolution de problèmes de consistance utilisant la contrainte souple SSE en conjonction avec des contraintes dures, peuvent être significativement améliorées. Par exemple, un algorithme d'établissement d'un niveau de consistance en (surclasse, x) pour un système de contraintes basé sur les préférences ($\mathcal{R}_0$, $\{c_{\mathrm{SSE}}[\alpha, z]\}_{\alpha, z}$), dont la contrainte souple finale (§ 5.3.2) est la contrainte souple de Suppes-Sen, peut être conçu à partir de l'Algorithme 10 en le modifiant très légèrement. Il suffit de filtrer les contraintes individuelles en même temps que les contraintes du système de contraintes. L'Algorithme 10 est alors corrigé en rectifiant son entête et en initialisant C non plus à l'ensemble vide mais à l'ensemble des contraintes de $\mathcal{R}_0$: L'instruction '$C \leftarrow \varnothing$' est remplacée par l'instruction '$C \leftarrow C(\mathcal{R}_0)$', avec $\mathcal{R}_0 = (V, D_0, C(\mathcal{R}_0))$.

```
FiltreDFS_SSE{α}(I_e : ensemble d'individus, V : ensemble de variables,
                 D_0 : ensemble de domaines, {c≥[α, y, i, j]}_α, y, i, j : contrainte souple
                 étendue, x : solution) : retourner {ensemble de domaines, non}
début
  i, k, l ∈ variables basées sur un indexage de I_e, avec card(I_e) = n.
  ℛ = (V, Dc, C) : {système de contraintes, non} ← non   vaut "non" s'il n'existe pas de système
                                                         de contraintes courant consistant.
  Ds : {ensemble de domaines indexés sur V, non} ← non   domaines courants des valeurs con-
                                                         sistantes (pour lesquelles, il existe un couplage parfait).
  D_PM : ensemble de domaines indexés sur I_e            domaines des variables d'itérations.
  valeur(·) : vecteur de n composantes entières          représente le couplage courant.
  trouvé : booléen   vaut "non" s'il est nécessaire de retourner en arrière.
  k ← 1   niveau courant de l'arborescence pour lequel on cherche une arête dont l'une des extrémités est u_1(k)
  C ← ∅
  tant que ( k ≤ n ) faire
    D_PM(k) ← I_e \ {valeur(i) pour tout i < k}   initialisation des valeurs possibles pour k.
    l ← choisir(D_PM(k))   choisit une valeur dans D_PM(k).
    trouvé ← non   le sommet courant u_1(k) n'a pas encore été couplé
    tant que ( D_PM(k) ≠ ∅ et trouvé = non ) faire
      D_PM(k) ← D_PM(k) \ {l}   la valeur courante est supprimée de D_PM(k)
      C ← C ∪ {c≥[α, x, k, l]}   ajout de la contrainte individuelle candidate
      ℛ ← filtrer((V, D_0, C))   test d'un niveau de consistance adaptable donné par filtrer.
      si ( ℛ ≠ non et k < n ) alors   le sommet courant u_1(k) a été couplé au sommet u_2(l) avec
        trouvé ← oui   succès, mais le couplage obtenu n'est toujours pas parfait.
      sinon
        C ← C \ {c≥[α, x, k, l]}
        si ( ℛ ≠ non et k = n ) alors   avec ℛ = (V, Dc, C)
          valeur(n) ← l   u_1(n) a été couplé à u_2(l) avec succès, aboutissant ainsi à un couplage
          Ds ←[188] Ds ∪ Dc   parfait consistant : M = {(u_1(i), u_2(valeur(i))) ∀ i ≤ n}
        sinon   le système de contraintes courant est inconsistant
          l ← choisir(D_PM(k))   une nouvelle valeur candidate est choisie.
        fin si
      fin si
    fin tant que
    si ( trouvé = oui ) alors   valeur(k) ← l ;   k ← k + 1
                                ici, l'ensemble {(u_1(i), u_2(valeur(i))) pour tout i < k} forme un couplage consis-
                                tant. On essaie alors d'ajouter une arête supplémentaire.
    sinon   le couplage actuel est inconsistant. On retourne en arrière.
      k ← k – 1
      si ( k = 0 ) alors   retourner Ds   fin si   toute l'arborescence a été parcourue
    fin si
  fin tant que
  retourner Ds
fin FiltreDFS_SSE{α}
```

Algorithme 10 : Algorithme de filtrage adaptable global de la contrainte SSE en (α, *x*) avec α ∈ {indifférent, surclasse, sousclasse}.

[188] Pour plus de clarté, nous écrivons abusivement $Ds \leftarrow Ds \cup Dc$ pour :
si ($Ds \neq$ non) **alors** $Ds(v) \leftarrow Ds(v) \cup Dc(v)$, pour tout $v \in V$ **sinon** $Ds \leftarrow Dc$ **fin si**

7.4.5.2. Filtrage Global Adaptable pour la Préférence Stricte et l'Aversion Stricte

Étudions maintenant le cas de la contrainte SSE en (mieux, x). Sa sémantique est la suivante :

$$S(c_{\text{SSE}}[\text{mieux}, x], V, D) = \{y \in \mathcal{D}_V \text{ tel que : } \exists \text{ une permutation } \sigma \text{ de } I_e \text{ avec } \forall\, i \in I_e, (y, i) \succcurlyeq (x, \sigma(i)) \text{ et qu'} \exists\, j \in I_e \text{ vérifiant } (y, j) \succ (x, \sigma(j))\}.$$

Une approximation peut être obtenue en utilisant les couvertures de sommets de graphes. Un résultat équivalent à la Proposition 25 est obtenu, en remplaçant la formule (7–48) par la suivante :

$$\begin{aligned} S(c_{\text{SSE}}[\text{mieux}, x], V, D) \subseteq & \left(\bigcap_{i \in I_e} \bigcup_{j \in I_e} S(c_{\succcurlyeq}[\text{surclasse}, x, i, j], V, D)\right) \cap \\ & \left(\bigcap_{j \in I_e} \bigcup_{i \in I_e} S(c_{\succcurlyeq}[\text{surclasse}, x, i, j], V, D)\right) \cap \\ & \left(\bigcup_{j \in I_e} \bigcup_{i \in I_e} S(c_{\succcurlyeq}[\text{mieux}, x, i, j], V, D)\right). \end{aligned} \tag{7–51}$$

Par rapport à la formule (7–48) en (surclasse, x), les deux premiers termes du second membre de (7–51) sont équivalents au second membre de (7–48). Ceci va de soit, sachant que $S(c_{\text{SSE}}[\text{mieux}, x], V, D) \subseteq S(c_{\text{SSE}}[\text{surclasse}, x], V, D)$. Le troisième terme indique qu'il est nécessaire qu'au moins un couple $(i, j) \in I_e \times I_e$ identifie une solution y de $S(c_{\succcurlyeq}[\text{mieux}, x, i, j], V, D)$, qui soit réalisable pour $S(c_{\text{SSE}}[\text{surclasse}, x], V, D)$. Ce terme est une approximation de la partie existentielle de la sémantique de $S(c_{\text{SSE}}[\text{mieux}, x], V, D)$.

Maintenant, par rapport aux couplages parfaits, que pouvons-nous dire ? La définition de $S(c_{\text{SSE}}[\text{mieux}, x], V, D)$ indique que pour chaque solution réalisable, il existe un couplage parfait dans son graphe biparti en (surclasse, x), et qu'au moins une des arêtes du couplage identifie une contrainte individuelle en (mieux, x) contenant cette solution réalisable. Formellement cela donne :

$$\begin{aligned} S(c_{\text{SSE}}[\text{mieux}, x], V, D) = \{y \in \mathcal{D}_V \text{ tel que : } & P_{PM}(G_{\text{surclasse}, x}(y)) \neq \varnothing \text{ et} \\ & \exists\, e \in A \text{ vérifiant : } y \in S(c_{\succcurlyeq}[\text{mieux}, x, i, j], V, D)\} \end{aligned} \tag{7–52}$$

avec A l'ensemble des arêtes du graphe $G_{\text{surclasse}, x}(y)$, et l'arête $e = (u_1(i), u_2(j))$.

Cette formule est pour la contrainte SSE en (mieux, x) homologue à la formule (7–49) pour la contrainte SSE en (α, x) avec α = {indifférent, surclasse, sousclasse}. Elle exprime la sémantique de la contrainte SSE en (mieux, x) au moyen des couplages. Nous obtenons alors directement l'équivalent de la Propriété 8 pour la préférence stricte :

Propriété 9 :

$$S(c_{\text{SSE}}[\text{mieux}, x], V, D) = \bigcup_{M \in PM(SSE, \text{surclasse}, x, n)} \left(\left(\bigcap_{\substack{(i,j) \in I_e^2 \text{ tel que :} \\ (u_1(i), u_2(j)) \in M}} S(c_{\succcurlyeq}[\text{surclasse}, x, i, j], V, D) \right) \cap \left(\bigcup_{\substack{(i,j) \in I_e^2 \text{ tel que :} \\ (u_1(i), u_2(j)) \in M}} S(c_{\succcurlyeq}[\text{mieux}, x, i, j], V, D) \right) \right)$$

Cette propriété se déduit directement de (7–52). Par rapport à la Propriété 8, le second terme du second membre est ajouté pour tenir compte de la satisfaction d'au moins une contrainte individuelle en (mieux, x). Elle suggère, comme au § 7.4.5.1 d'utiliser l'énumération des couplages parfaits valides en (surclasse, x) pour SSE, pour concevoir des algorithmes de filtrage adaptables pour la contrainte $c_{\text{SSE}}[\text{mieux}, x]$. Une manière de procéder est de tester, dès qu'un

couplage parfait valide est obtenu, si au moins une des arêtes identifie une contrainte individuelle qui, ajoutée aux autres contraintes individuelles, définit un système de contraintes consistant.

Dans cet ordre d'idées, l'Algorithme 10 en (surclasse, x), c'est-à-dire `FiltreDFS_SSE{surclasse}`, peut être modifié pour définir un algorithme de filtrage adaptable pour c_{SSE}[mieux, x]. Il suffit alors de remplacer l'instruction '$Ds \leftarrow^{188} Ds \cup Dc$' par l'instruction :

$$Ds \leftarrow^{188} Ds \cup \texttt{TCP_SSE}\{\text{mieux}\}(I_e, \mathcal{R}, valeur, \{c_{\succcurlyeq}[\alpha, y, i, j]\}_{(\alpha, y, i, j)}, x)$$

avec `TCP_SSE{mieux}` un sous-programme dont la description est donnée ci-dessous :

`TCP_SSE{mieux}`(I_e : ensemble d'individus, $\mathcal{R} = (V, D_0, C)$: système de contraintes, $valeur(\cdot)$: vecteur d'entiers, $\{c_{\succcurlyeq}[\alpha, y, i, j]\}_{(\alpha, y, i, j)}$: contrainte souple étendue, x : solution) : **retourner** {ensemble de domaines, non}

début

$i \in \quad \leftarrow 1$ — variable de boucle

$\mathcal{R}_1 = (V, Dc, C)$: {système de contraintes, non} $\leftarrow$ non — vaut "non" s'il n'existe pas de système de contraintes courant consistant.

Ds : {ensemble de domaines indexés sur V, non} $\leftarrow$ non — domaines courants des valeurs consistantes.

pour tout ($i = 1$ **à** n)

$C \leftarrow C \cup \{c_{\succcurlyeq}[\text{mieux}, x, i, valeur(i)]\}$ — ajout de la contrainte individuelle candidate

$\mathcal{R}_1 \leftarrow$ <u>filtrer</u>$((V, D_0, C))$ — test d'un niveau de consistance adaptable donné par <u>filtrer</u>.

si ($\mathcal{R}_1 \neq$ non) **alors**

$Ds \leftarrow^{188} Ds \cup Dc$ — la contrainte individuelle a laissé le système de contraintes consistant.

fin si

$C \leftarrow C \setminus \{c_{\succcurlyeq}[\text{mieux}, x, i, valeur(i)]\}$

fin pour

retourner Ds

fin `TCP_SSE{mieux}`

Algorithme 11 : Sous-programme du filtrage adaptable de la contrainte SSE en (mieux, x).

Tout ce qui vient d'être exposé pour la préférence stricte est valable pour l'aversion stricte, en remplaçant toute occurrence de l'attitude 'mieux' par 'pire' et 'surclasse' par 'sousclasse'.

7.4.5.3. Filtrage Adaptable pour l'Incomparabilité

Clôturons cette section avec la contrainte SSE en (incomparable, x). Sa sémantique est la suivante :

$$\begin{aligned} S(c_{\text{SSE}}[\text{incomparable}, x], V, D) = \{ & y \in \mathcal{D}_V \text{ tel que : toute permutation } \sigma \text{ de } I_e \text{ vérifie que :} \\ & \exists\, i \in I_e, \text{non}[(y, i) \succcurlyeq (x, \sigma(i))] \text{ et non}[(y, i) \preccurlyeq (x, \sigma(i))]\} \\ = \{ & y \in \mathcal{D}_V \text{ tel que : toute permutation } \sigma \text{ de } I_e \text{ vérifie que :} \\ & \exists\, i \in I_e \text{ avec } (y, i) \parallel (x, \sigma(i)) \text{ ou} \\ & \exists\, (i, j) \in I_e^2, i \neq j \text{ avec } (y, i) \succ (x, \sigma(i)) \text{ et } (y, j) \prec (x, \sigma(j))\} \end{aligned}$$

La formulation de cette contrainte sous forme d'unions et d'intersections de contraintes individuelles est alors directe :

$$S(c_{\mathrm{SSE}}[\text{incomparable}, x], V, D) = \left(\bigcup_{(i,j)\in I_e^2} S(\, c_{\succcurlyeq}[\text{incomparable}, x, i, j], V, D \,) \right) \cup$$

$$\left(\bigcup_{(i,j)\in I_e^2} \bigcup_{k\in I_e\setminus\{i\}} \bigcup_{l\in I_e\setminus\{j\}} S(c_{\succcurlyeq}[\text{mieux}, x, i, j], V, D) \cap S(c_{\succcurlyeq}[\text{pire}, x, k, l], V, D \,) \right)$$

Le second membre est formé d'un terme de n^2 contraintes individuelles, et d'un terme de $n^2(n - 1)^2$ de conjonction de deux contraintes individuelles. À partir de cette formule, des algorithmes de filtrage adaptables peuvent être conçus.

8. Application sur un PEPTAG : Résultats et Performances

8.1. Variables et Contraintes du Prototype

Pour tester l'efficacité algorithmique de certains des algorithmes présentés au chapitre précédent, nous avons développé un prototype ciblé sur l'élaboration de plannings infirmiers en utilisant la programmation par contraintes. Les exemples traités sont des PEPTAG, dont le modèle de faisabilité est un système de contraintes $\mathcal{R} = (V, D, C)$ basé sur la dichotomie $(E, J \mid EV)$, avec E un ensemble d'infirmiers, J un ensemble de jours consécutifs, et EV un ensemble d'étiquettes. En d'autres termes, une variable $v(e, j)$ est définie pour chaque couple (employé, jour) $(e, j) \in E \times J$, le domaine de chacune de ces variables étant initialisé à l'ensemble des étiquettes.

Les contraintes génériques utilisées pour définir les exemples sont les suivantes :

- des contraintes de charges,
- des contraintes d'affectation ponctuelle,
- des contraintes de changement d'étiquettes,
- des contraintes de fréquence des étiquettes,
- des contraintes de disponibilité des agents,
- des contraintes de séquences d'étiquettes.

Voici leur description :

8.1.1. Contrainte d'Affectation Ponctuelle

Forme Déclarative : Le jour ***j*** l'agent ***e*** est **toujours** (resp. **jamais**) affecté à l'une des étiquettes de l'ensemble $\boldsymbol{EV_1} \subseteq EV$.

Exemples :

- l'agent Sidonie sera [obligatoirement] au repos le 14 juillet 1997.
- l'agent Lolita sera [obligatoirement] soit de Nuit, soit de Soir le 20 juillet 1997.

Notations et Remarques : Cette contrainte est la plus simple. Elle est notée $c_{ap}(e, j, EV_1)$ lorsque la forme affirmative (adverbe "toujours") est utilisée, et est notée $c_{ap}(e, j, EV \setminus EV_1)$ lorsque la forme négative (adverbe "jamais") est utilisée.

Définition de la Contrainte : Contrainte unaire sur la variable $v(e, j)$ qui limite le domaine au sous-ensemble d'étiquettes $EV_1 \subseteq EV$: $S(c_{ap}(e, j, EV_1)) = EV_1 \cap D(v(e, j))$.

8.1.2. Contrainte de Charges

Ce type de contraintes traduit la demande en agent pour chaque étiquette de la journée. Avant de générer le planning, on connaît la demande pour chaque jour de l'horizon de planification en chaque étiquette.

Forme Déclarative : Le jour ***j*** les besoins sont d'au moins ***l*(*ev*)** , au plus ***u*(*ev*)** agents pour l'étiquette ***ev***.

Notation : $c_{chg}(j, l, u)$, avec $j \in J$, l et u deux vecteurs entiers indexés par les étiquettes EV.

Définition de la Contrainte : Une contrainte de charges $c_{chg}(j, l, u)$ sur un jour j est une contrainte pour laquelle chaque étiquette ev est associée à deux entiers positifs $l(ev)$ et $u(ev)$

traduisant respectivement le nombre minimum et le nombre maximum d'agents requis pour cette étiquette.

$c_{chg}(j, l, u)$ porte sur toutes les variables du jour j : $V(c_{chg}(j, l, u)) = \{v(e, j)$ pour tout $e \in E\}$, et son ensemble de solutions réalisables est :

$S(c_{chg}(j, l, u)) = \{x \in \mathcal{D}_{V(chg)}$ tel que : $\forall\ ev \in EV$:

$$l(ev) \leq \text{card}(\{v \in V(c_{chg}(j, l, u)) \mid x(v) = ev\}) \leq u(ev)\ \}$$

avec $\mathcal{D}_{V(chg)} = \underset{v \in V(c_{chg}(j,l,u))}{\times} D(v)$, et

$\text{card}(\{v \in V(c_{chg}(j, l, u)) \mid x(v) = ev\})$ = le nombre de fois ev apparaît dans le tuple x.

8.1.3. Contrainte de Changement d'Étiquettes

Une contrainte de changement d'étiquettes, ou contrainte de transition, est une contrainte générique faisant intervenir un agent e de E, un couple de jours j_α et j_β, un couple de jours de semaine $js_1, js_2 \in$ {dimanche, ..., samedi}, et un ensemble de couples d'étiquettes $EVEV \subset EV \times EV$.

Forme Déclarative : Pour un agent $\boldsymbol{e}$, l'étiquette $\boldsymbol{ev}$, le jour de semaine $\boldsymbol{js_1}$, **n'**est **jamais** (resp. est **toujours**) suivie des étiquettes de $\boldsymbol{EV_1} \subseteq EV$, le jour de semaine $\boldsymbol{js_2}$, de la date $\boldsymbol{j_\alpha}$ à la date $\boldsymbol{j_\beta}$.

Précisons que l'ensemble EV_1 est défini de manière à ce qu'il vérifie : $\{ev\} \times EV_1 \subseteq EVEV$ et $EVEV \setminus (\{ev\} \times EV_1)$ ne contient aucun couple (ev_1, ev_2) avec $ev_1 = ev$.

Exemples :

- Pour tous, jamais Nuit est suivi de Matin ou de Soir du 01 / 01 / 1997 au 31 / 12 / 1999.
- Un agent travaillant le mercredi, ne travaille pas le samedi suivant.

Notations et Remarques : Cette contrainte est notée $c_{ce}(e, j_\alpha, j_\beta, js_1, js_2, EVEV)$ lorsque la forme positive (adverbe "toujours") est utilisée, et est notée $c_{ce}(e, j_\alpha, j_\beta, js_1, js_2, (EV \times EV) \setminus EVEV)$ lorsque la forme négative (adverbe "jamais") est utilisée. Remarquons que l'ordre des jours de semaine js_1 et js_2 est important.

Définition de la Contrainte : Une contrainte de changement d'étiquettes porte sur les variables :

$V(c_{ce}(e, j_\alpha, j_\beta, js_1, js_2, EVEV)) = \{v(e, j)$ telles que : $j_\alpha \leq j \leq j_\beta$, $\text{jour_semaine}(j) \in \{js_1, js_2\}$, $\text{jour_semaine}(j_{\min}) = js_1$ et $\text{jour_semaine}(j_{\max}) = js_2\}$

avec : $j_{\min} = \min\{j$ tel que $v(e, j) \in V(c_{ce}(e, j_\alpha, j_\beta, js_1, js_2, EVEV))\}$ et

$j_{\max} = \max\{j$ tel que $v(e, j) \in V(c_{ce}(e, j_\alpha, j_\beta, js_1, js_2, EVEV))\}$

Son ensemble de solutions réalisables est :

$S(c_{ce}(e, j_\alpha, j_\beta, js_1, js_2, EVEV)) = \{x \in \mathcal{D}_{V(ce)}$ tels que : $\forall\ k \in \{1, \ldots, p-1\}$, avec $\text{jour_semaine}(j_{2k-1}) = js_1$ et $\text{jour_semaine}(j_{2k}) = js_2$, alors : $(x(v(e, j_{2k-1})), x(v(e, j_{2k}))) \in EVEV\ \}$.

avec $\mathcal{D}_{V(ce)} = \underset{v \in V(c_{ce}(e, j_\alpha, j_\beta, js_1, js_2, EVEV))}{\times} D(v)$,

$2p = \text{card}(\ V(c_{ce}(e, j_\alpha, j_\beta, js_1, js_2, EVEV))\ )$ et $j_{\min} = j_1 < j_2 < \ldots < j_{2p} = j_{\max}$.

8.1.4. Contrainte de Fréquence des Étiquettes

Forme Déclarative : Pour l'agent $\boldsymbol{e}$, il faut que l'étiquette $\boldsymbol{ev}$ apparaisse au moins $\boldsymbol{l(ev)}$ et au plus $\boldsymbol{u(ev)}$ fois tous les $\boldsymbol{f}$ jours sur la période commençant le jour $\boldsymbol{j_1}$ et finissant le jour $\boldsymbol{j_2}$.

Exemples :

- Chaque agent a entre 4 et 6 jours de repos toutes les 2 semaines, du 01 / 01 / 1997 au 31 / 12 / 1999.
- L'agent Marthe travaille exactement 7 nuits toutes les 2 semaines, du 01 / 01 / 1997 au 31 / 12 / 1999.

Notation : Une contrainte de fréquence d'étiquettes est une contrainte générique $c_{fe}(e, j_1, j_2, f, EV_1, l, u)$ associée à un agent e, un couple de jours j_1 et j_2, un ensemble d'étiquettes $EV_1 \subseteq EV$, chaque étiquette ev de EV_1 étant associée à deux entiers positifs $l(ev)$ et $u(ev)$.

Définition de la Contrainte : Une contrainte de fréquence des étiquettes porte sur les variables : $V(c_{fe}(e, j_1, j_2, f, EV_1, l, u)) = \{v(e, j)$ telles que $j_1 \leq j \leq j_2\}$

et admet pour ensemble de solutions réalisables :

$S(c_{fe}(e, j_1, j_2, f, EV_1, l, u)) = \{x \in \mathcal{D}_{V(fe)}$ tels que : pour la partition $\{S_1, ..., S_{q+1}\}$ de $V(c_{fe}(e, j_1, j_2, f, EV_1, l, u))$ en séquences de f variables consécutives :

$\forall$ ev étiquette de EV et $\forall$ i, $1 \leq i \leq q + 1$, on a

$l(ev) \leq \text{card}(\{v \in S_i \mid x(v) = ev\} \leq u(ev)\}$

avec :

- $\mathcal{D}_{V(fe)} = \underset{v \in V(c_{ce}(e, j_1, j_2, EVEV))}{\times} D(v)$;
- La partition $\{S_1, ..., S_{q+1}\}$ de $V(c_{fe}(e, j_1, j_2, f, EV_1, l, u))$ est définie comme suit :
  Si card($V(c_{fe}(e, j_1, j_2, f, EV_1, l, u))$) $= q.f + r$ avec, q et r deux entiers positifs et $r < f$, alors on définit $\{S_1, ..., S_{q+1}\}$ une partition de $V(c_{fe}(e, j_1, j_2, f, EV_1, l, u))$ en regroupant :
  . dans S_i, $1 \leq i \leq q$ les variables dont l'indice j varie de $j_1 + f.(i - 1)$ à $j_1 + f.i - 1$, et
  . dans S_{q+1} les variables dont l'indice varie de $j_1 + f.q$ à $j_1 + f.q + r = j_2$.
- card($\{v \in S_i \mid x(v) = ev\}$) = le nombre de fois que la valeur ev est assignée à une variable de S_i dans le tuple x de $\mathcal{D}_{V(fe)}$;

8.1.5. Contrainte de Disponibilité des Agents

Forme Déclarative : L'agent ***e*** est **toujours** (ou **jamais**) affecté à l'étiquette ***ev*** pendant les jours de semaine de la liste ***JS*** $\subseteq$ {dimanche, ..., samedi} entre ***l(ev)*** et ***u(ev)*** fois sur ***f*** jours, de la date $\boldsymbol{j_1}$ à la date $\boldsymbol{j_2}$ incluse.

Exemples :

- l'agent John n'est jamais affecté au matin les mercredi, du 01 / 01 / 1997 au 31/ 12 / 1999,
- l'agent Cédric est au repos un dimanche sur 2, du 01 / 01 / 1997 au 31 / 12 / 1999.

Notation : Dans sa forme positive (adverbe "toujours") cette contrainte est notée $c_{da}(e, j_1, j_2, f, JS_1, EV_1, l, u)$, et dans sa forme négative (adverbe "jamais") elle est notée $c_{da}(e, j_1, j_2, f, JS_1, EV \setminus EV_1, l, u)$.

Les paramètres de cette contrainte générique sont : un employé e, un couple de jours j_1 et d_2, une fréquence f, un ensemble de jours de semaines $JS \subseteq$ {dimanche, ..., samedi}, un ensemble d'étiquettes $EV_1 \subseteq EV$, chaque étiquette ev de EV_1 étant associée à deux entiers positifs $l(ev)$ et $u(ev)$.

Définition de la Contrainte : Une contrainte de disponibilité des agents porte sur les variables :

$V(c_{da}(e, j_1, j_2, f, JS_1, EV_1, l, u)) = \{x(e, j)$ telles que $j_1 \leq j \leq j_2$ et jour_semaine(j) $\in JS\}$.

C'est-à-dire que la contrainte porte uniquement sur les jours de semaines de JS compris entre le jour j_1 et le jour j_2.

Considérons la partition $\{S_1, ..., S_{q+1}\}$ de $V(c_{da}(e, j_1, j_2, f, JS_1, EV_1, l, u))$ en séquences de $f \cdot \frac{|JS_1|}{7}$ variables consécutives. Alors l'ensemble des solutions réalisables de la contrainte de disponibilité des agents est :

$S(c_{da}(e, j_1, j_2, f, JS_1, EV_1, l, u)) = \{x \in \mathcal{D}_{V(da)}$ tels que : $\forall$ ev étiquette de EV_1 et $\forall$ i, $1 \leq i \leq q + 1$, on a $l(ev) \leq \text{card}(\{v \in S_i \mid x(v) = ev\}) \leq u(ev)\}$

avec $\mathcal{D}_{V(da)} = \underset{v \in V(c_{da}(e, j_1, j_2, f, JS_1, EV_1, l, u))}{\times} D(v)$.

8.1.6. Contrainte de Séquences d'Étiquettes

Forme Déclarative : L'agent $\boldsymbol{e}$ n'est jamais affecté aux étiquettes de $\boldsymbol{EV_1}$ plus de $\boldsymbol{u}$ fois de suite durant la période allant du jour $\boldsymbol{j_1}$ au jour $\boldsymbol{j_2}$ inclus.

S'il effectue exactement $\boldsymbol{u}$ fois les étiquettes de $\boldsymbol{EV_1}$, alors il faut lui affecter des étiquettes de $\boldsymbol{EV_2}$ $\boldsymbol{n}$ fois de suite.

Exemple : aucun agent ne travaille plus de 4 nuits successives, du 01 / 01 / 1997 au 31 / 12 / 1999. S'il travaille 4 nuits consécutives, alors il lui faut au moins 2 jours de repos consécutifs tout de suite après les nuits.

Notation : Cette contrainte est notée $c_{se}(e, j_1, j_2, EV_1, EV_2, u, n)$, avec e un employé, j_1 et j_2 un couple de jours, EV_1, EV_2 des sous-ensembles d'étiquettes avec $EV_1 \cap EV_2 = \emptyset$, et u et n deux entiers positifs.

Définition de la Contrainte : Une contrainte de séquence d'étiquettes porte sur les variables :

$$V(c_{se}(e, j_1, j_2, EV_1, EV_2, u, n)) = \{x(e, j) \text{ telles que } j_1 \leq j \leq j_2\}$$

et admet l'ensemble des solutions réalisables suivant :

$S(c_{se}(e, j_1, j_2, EV_1, EV_2, u, n)) = \{x \in \mathcal{D}_{V(se)}$ tels que : pour toute séquence S de $(u + 1)$ variables consécutives de $V(c_{se}(e, j_1, j_2, EV_1, EV_2, u, n))$:

$$\mathsf{card}(\{v \in S \mid x(v) \in EV_1\}) \leq u,$$

et pour toute séquence $S' = (S_u + S_n)$ de $(u + n)$ variables consécutives de $V(c_{se}(e, j_1, j_2, EV_1, EV_2, u, n))$ (S_u = séquence des u premières variables de S', S_n les n suivantes) :

$$\text{si } \mathsf{card}(\{v \in S_u \mid x(v) \in EV_1\}) = u \text{ alors } \mathsf{card}(\{v \in S_n \mid x(v) \in EV_2\}) = n \}$$

avec $\mathcal{D}_{V(se)} = \bigtimes_{v \in V(c_{se}(e, j_1, j_2, EV_1, EV_2, u, n))} D(v)$.

Selon les différentes combinaisons de valeurs de paramètres des contraintes génériques utilisées, différentes instances de PEPTAG sont obtenues. Ce problème de satisfaction de contraintes a donné lieu à un prototype réalisé dans le cadre d'un accord avec l'entreprise Equitime SA (voir JOSEPH [1998]).

Ce prototype a été complété pour aboutir à un nouveau prototype traitant de préférences dans les problèmes de planification infirmière.

8.2. Modèle de Préférence

Nous avons décidé de ne traiter que des exemples cherchant à satisfaire une demande exacte en effectif pour chaque couple (jour, étiquette). Ce qui implique une variation des coûts négligeable, et une variation de la qualité de service supposée entièrement dépendante de la qualité de vie des employés. De plus, nous ne considérons qu'un vrai-critère étendu comme préférences individuelles. Ce vrai-critère étendu exprime le nombre de séquences de f repos à maximiser pour chaque individu. Il a été modélisé par $\mathsf{card}(E)$ contraintes formant une VC-contrainte souple étendue (§ 5.2.3). Comme nous l'avons souligné, tout vrai-critère étendu sur un ensemble d'individus observés E est équivalent à un profil de $\mathsf{card}(E)$ vrai-critères simples de même dimension.

Par conséquent, $\mathsf{card}(E)$ nouvelles variables valeurs $V_{\#\text{repos}}$ ont été introduites, ainsi que les domaines associés. Chaque domaine $D(v_{\#}(e))$ de la variable $v_{\#}(e) \in V_{\#\text{repos}}$, est un sous-ensemble d'entiers positifs inclus dans l'intervalle entier $[0, \mathsf{card}(J) - f - 1]$. En d'autres termes, le nombre de séquences de repos dans un horaire varie de 0 (il n'existe pas de séquences d'au moins f repos dans l'horaire considéré pour e) à $\mathsf{card}(J) - f - 1$ (e est au repos durant tout l'horizon de planification).

8.2.1. Contrainte de Dénombrement de Séquences d'Étiquettes

La VC-contrainte souple étendue a été implémentée par l'intermédiaire d'une contrainte générique appelée contrainte de dénombrement de séquences d'étiquettes.

Forme Déclarative : Le nombre de séquences de ***f*** étiquettes ***ev*** contenu dans l'horaire de travail de l'agent ***e*** est un élément de $\boldsymbol{D(v_{\#}(e))}$.

Notation : Notée $c_{\#se}(e, f, ev, v_{\#}(e))$, la contrainte de dénombrement de séquences d'étiquettes reçoit en paramètres un employé e, un entier naturel f, une étiquette ev, et la variable valeur $v_{\#}(e)$ associée à e et au vrai critère simple considéré.

Exemple : Si l'on prend EV = {T, R} et (f, ev) = (2, R) alors la suite d'étiquettes RRTTTTRTTRRRTT contient 3 séquences de 2 étiquettes R.

Définition de la Contrainte : Une contrainte de dénombrement de séquences d'étiquettes porte sur les variables : $V(c_{\#}(e, f, ev, v_{\#}(e))) = \{x(e, j)$ telles que $j \in J\} \cup \{v_{\#}(e)\}$,
et admet l'ensemble des solutions réalisables suivant :
$S(c_{\#}(e, f, ev, v_{\#}(e))) = \{x \in \mathcal{D}_{V(\#)}$ tels que : le nombre de séquences S de f variables consécutives de $V(c_{\#}(e, f, ev, v_{\#}(e))) \setminus \{v_{\#}(e)\}$ instanciées à l'étiquette ev, est un entier de $D(v_{\#}(e))$ }
avec $\mathcal{D}_{V(\#)} = \bigtimes_{v \in V(c_{\#}(e,f,ev,v_{\#}(e)))} D(v)$.

8.2.2. Agrégation par la Dominance de Pareto

Nous nous sommes limités à l'agrégation de ces préférences individuelles par la dominance de Pareto (§ 7.4.1). Elle est appliquée sur les variables valeurs $V_{\#repos}$ de la VC-contrainte souple étendue.

Ce modèle préférentiel est très limité par rapport à toute la théorie qui a été présentée. Par contre, il constitue un premier pas vers l'utilisation du multicritère en programmation par contraintes, et l'une des rares applications du problème d'élaboration de plannings infirmiers basées sur la recherche de solutions maximales au sens de Pareto et utilisant uniquement la consistance.

8.3. Résolution

8.3.1. Modes de Résolution

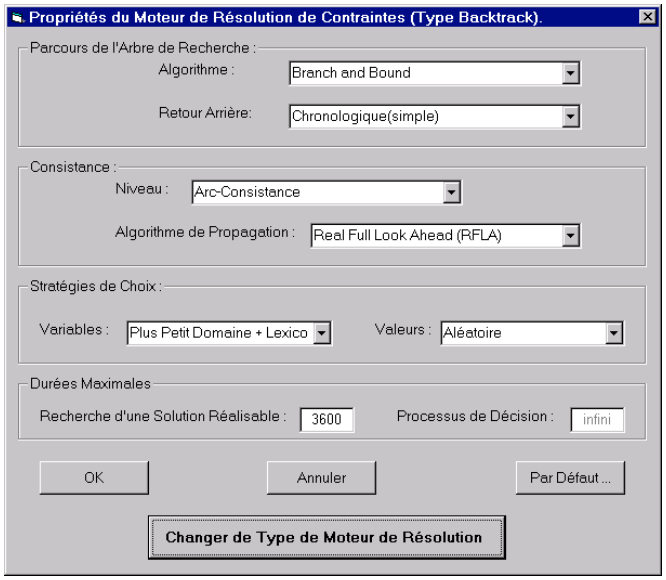
Plusieurs types de résolution sont proposés aux utilisateurs. Une première basée sur le maintien d'un niveau de consistance, une seconde basée sur la recherche d'une solution réalisable et une troisième basée sur la recherche d'une solution maximale. Détaillons :

Le processus de maintien d'un niveau de consistance réagit aux modifications de domaines effectuées par les utilisateurs en supprimant les valeurs inconsistantes pour l'arc-consistance (AC) ou/et en ajoutant les valeurs qui ne sont plus inconsistantes pour AC. En cas d'incohérence, une gestion de celle-ci est proposée en apportant des indications sur une combinaison d'affectations, de domaines et de contraintes impliquant une incohérence. Leur relaxation est proposée. Cette proposition est effectuée jusqu'à ce qu'un système de contraintes arc-consistant soit obtenu.

La recherche d'une solution réalisable se fait par backtracking (BT) et AC, retournant un planning réalisable, sinon "non".

La recherche de solution Pareto-optimale, retournant un planning maximal pour le modèle préférentiel, si un tel planning existe. Sinon retourne "non".

Il est possible de combiner les différents modes de résolution.



Exemple de paramètres du moteur de résolution :

- stratégie de choix d'une variable : plus petit domaine + lexicographique
- stratégie de choix d'une valeur : aléatoire
- stratégie de retour arrière : simple
- niveau de consistance : arc-consistance
- algorithme de propagation : le real-full-lookahead.

Figure 54 : Interface utilisé pour paramétrer la résolution.

8.3.2. Stratégies d'Évaluation

Les trois modes de résolution sont basés sur la programmation par contraintes. Dans ce cadre, un algorithme de filtrage a été conçu pour chaque contrainte générique dure ou souple. Ainsi, pour la résolution arborescente, les stratégies d'évaluation implémentées dans évaluer(J) testent la consistance du système de contraintes J courant et établissent un niveau de consistance sur J.

8.3.2.1. Niveaux de Consistance

Les niveaux de consistance utilisés sont basés sur l'arc-consistance (AC) et sont :

- le real-full-lookhead (RFLA),
- le forward-checking (FC) et
- une stratégie effectuant un tirage aléatoire entre les deux niveaux précédents à chaque itération. Dans ce dernier cas, plusieurs probabilités de tirage ont été testées :
  Si on note $p \in [0, 1]$ la probabilité de tirage de RFLA, alors la stratégie mixte est notée : p·RFLA + $(1 - p)$·FC.

Un algorithme de filtrage a donc été conçu pour chaque contrainte globale. Nous détaillons succinctement ci-dessous :

8.3.2.2. Filtrage pour les Contraintes Dures

La contrainte de charge $c_{chg}(j, l, u)$ est équivalente à la contrainte globale de cardinalité (Annexe A1) suivante : $c_{\text{GCC}}(V(c_{chg}(j, l, u)), EV, D_j, [l, u])$. D_j correspond aux domaines associés aux variables $V(c_{chg}(j, l, u))$. Par conséquent, l'algorithme de RÉGIN [1994] établissant la GIC pour les contraintes GCC a été utilisé.

La contrainte de changement d'étiquettes $c_{ce}(e, j_\alpha, j_\beta, js_1, js_2, EVEV)$ est composée de contraintes binaires sur chaque couple de variables $(v(e, j_{2k-1}), v(e, j_{2k}))$ pour $k = 1, \ldots, p$ (en utilisant la notation du § 8.1.3). L'algorithme de filtrage implémenté gère le déclenchement du filtrage ses contraintes binaires.

La contrainte de fréquence des étiquettes $c_{fe}(e, j_1, j_2, f, EV_1, l, u)$ est équivalente à q+1 contraintes GCC disjointes, à raison d'une contrainte sur chaque séquence S_i de f variables (en utilisant la notation du § 8.1.4). La contrainte GCC associé à S_i reçoit comme paramètres le

quadruplet (S_i, EV, D_i, [l', u']) avec D_i les domaines associés à S_i, et [l', u'] le pavé de $\text{card}(EV)$

tel que $l'(ev) = f - \begin{cases} l(ev) & \text{si l'étiquette } ev \in EV_1 \\ f - \sum_{ev \in EV_1} u(ev) & \text{sinon} \end{cases}$ et $u'(ev) = \begin{cases} u(ev) & \text{si l'étiquette } ev \in EV_1 \\ f - \sum_{ev \in EV_1} l(ev) & \text{sinon} \end{cases}$.

La contrainte de disponibilité des agents $c_{da}(e, j_1, j_2, f, JS_1, EV_1, l, u)$ est, comme la contrainte de fréquence, équivalente à $q + 1$ contraintes GCC disjointes et reçoit les mêmes paramètres. La différence tient dans la définition des séquences de variables S_i.

L'algorithme de filtrage de la contrainte de séquences d'étiquettes $c_{se}(e, j_1, j_2, EV_1, EV_2, u, n)$ se décompose en 2 parties :

1. Toute séquence S_u de u variables consécutives, dont les domaines sont inclus dans EV_1, sont repérées, et les opérations suivantes sont effectuées :
   - toute occurrence d'étiquettes de EV_1 dans le domaine de la variable qui précède la séquence S_u est supprimée ;
   - les domaines des variables de la séquence S_n de n variables consécutives suivantes sont réduits aux étiquettes EV_2.
2. S'il existe des séquences de $(u + n + 1)$ variables telles que :
   - le domaine de la première variable ne contient aucune étiquette de EV_1 ;
   - les domaines des u variables suivantes S_u ont une intersection non vide avec EV_1 ;
   - il existe au moins une variable parmi les n dernières qui ne contient aucune étiquette de EV_2 ;

   alors la séquence de variables S_u contient au moins une variable dont le domaine ne contient aucune étiquette de EV_1.

8.3.2.3. Filtrage pour la VC-Contrainte Souple

L'algorithme de filtrage utilisé pour la contrainte de dénombrement de séquences d'étiquettes $c_{\#se}(e, f, ev, v_\#(e))$ est basé sur l'opérateur de cardinalité (§ 2.1.4.3). En effet, cette contrainte est équivalente à la contrainte $\#(v_\#(e), D(v_\#(e)), C)$, l'ensemble C étant formé des $\text{card}(J) - f + 1$ contraintes glissantes portant sur f variables consécutives de $\{v(e, j), \forall j \in J\}$. Une contrainte c de C est satisfaite si ses f variables sont instanciées à ev. Elle n'est pas satisfaite lorsque le domaine d'au moins une de ses variables ne contient pas ev.

Par conséquent, nous avons utilisé l'algorithme de filtrage pour l'opérateur de cardinalité proposé par VAN HENTENRYCK & DEVILLE [1991].

Le niveau de consistance obtenu est assez faible. Il peut être amélioré en utilisant des algorithmes de propagation plus spécialisés comme la contrainte globale de séquencement (RÉGIN & PUGET [1997]) ou encore la *cardinality-path constraint* de BELDICEANU [2000*b*].

8.3.3. Stratégies de Choix et de Séparation pour la Résolution Arborescente

Les algorithmes de décomposition arborescente très généraux présentés au § 7.1 ont été spécialisés pour résoudre le PEPTAG précédent. Ainsi, plusieurs stratégies de recherche ont été implémentées et testées sur des jeux de données (§ 8.4) pour identifier les plus efficaces.

Les stratégies de choix d'une sous-instance dans $\wp$, implémentées dans choisir($\wp$), sont toutes basées sur une recherche en profondeur, couplée à un retour arrière standard.

Les stratégies de séparation, divisant l'espace de recherche associé à J en morceaux plus petits, et implémentées dans générer sous-instances(J)), sont des instanciations statiques ou dynamiques des variables.

Ces deux stratégies ont été combinées pour aboutir à deux types de stratégies standards en programmation par contraintes : les stratégies de choix de variables et stratégies de choix de valeurs.

Ainsi, 5 stratégies de choix de variables ont été utilisées :

– l'ordre dynamique lexicographique sur les jours (JL) : A une itération donnée, la variable à instancier est choisie d'abord selon l'ordre croissant des jours (chronologique), puis selon un ordre prédéfini des agents ;
– l'ordre dynamique lexicographique sur les agents (AL) : Les rôles des agents et des jours sont inversés. La priorité est donnée à l'ordre prédéfinie des agents, puis parmi plusieurs variables le choix est fait selon l'ordre chronologique ;
– le choix aléatoire et uniforme des variables non instanciées (A) ;
– la stratégie dynamique du plus petit domaine, suivie de l'ordre chronologique (PPDJL) : A une itération donnée, la variable choisie est celle dont le domaine est le plus petit en nombre d'éléments, et parmi plusieurs candidats, la stratégie JL est appliquée ;
– la stratégie dynamique du plus petit domaine suivie du choix aléatoire (PPDA) : Le choix de la variable à instancier correspond à celle ayant un domaine de plus petite taille, et parmi plusieurs candidats, un tirage aléatoire uniforme est réalisé ;

Deux stratégies de choix de valeurs ont été implémentées : le choix lexicographique statique (L) et le choix aléatoire (A).

8.4. Expérimentations

8.4.1. Trois Exemples Traités

Nous avons expérimenté ces différents paramètres de résolution sur 3 exemples nommés PEPTAG4-14, PEPTAG8-14 et PEPTAG12-84. Leur description rapide est donnée dans le tableau suivant :

Nom	Nb d'agents	Nb jours	Nb d'étiquettes	Nb variables	Nb contraintes
PEPTAG4-14	4	14	5	56	22
PEPTAG8-14	8	14	4	112	80
PEPTAG12-84	12	84	4	1008	362

Tableau 8 : Description générale des 3 exemples de PEPTAG résolus.

Les 3 exemples respectent les contraintes dures suivantes :
– une seule étiquette par jour et par personne (contraintes d'affectation),
– une couverture exacte de la table de besoins (contraintes de charges) ; L'instance PEPTAG4-14 utilise la table de besoins décrit au Tableau 9 pour chaque semaine formant l'horizon de planification. L'instance PEPTAG8-14 utilise la table de besoins décrit au Tableau 10 ; et l'instance PEPTAG12-84 utilise la table de besoins décrit au Tableau 2 du § 2.2.1.1.

type vacation	L	M	M	J	V	S	D
J8	0	2	1	2	2	0	0
J9	0	2	1	1	1	0	0
Jlundi (Jl)	2	0	0	0	0	0	0
Jsamedi (Js)	0	0	0	0	0	2	0

Tableau 9 : Table de besoins de PEPTAG4-14.

type vacation	L	M	M	J	V	S	D	L	M	M	J	V	S	D
Matin	2	3	2	3	2	3	1	1	2	3	3	2	3	1
Soir	2	2	2	2	2	2	2	3	3	1	2	2	2	2
Nuit	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Tableau 10 : Table de besoins de PEPTAG8-14.

L'instance PEPTAG4-14 vérifie en plus les contraintes dures suivantes :
– pour tous exactement 6 repos sur l'horizon de planification :

$c_{fe}(e, 1, 14, 14, \{\text{repos}\}, l(\text{repos}) = 6, u(\text{repos}) = 6)$, pour tous les agents e.

– Un agent travaillant le samedi ne travaille pas le lundi suivant :

$c_{ce}(e, 1, 14, \text{samedi}, \text{lundi}, (\{\text{repos}\} \times EV) \cup (EV \times \{\text{repos}\}))$, pour tous les agents e.

L'instance PEPTAG8-14 vérifie en plus les contraintes dures suivantes :

- Les 6 premiers agents travaillent uniquement la journée :
  $c_{fe}(e_i, 1, 14, 14, \{\text{nuit}\}, u(\text{nuit}) = 0)$, pour tout $i \in \{1, \ldots, 6\}$
- Les 2 derniers agents travaillent uniquement la nuit :
  $c_{fe}(e_i, 1, 14, 14, \{\text{matin, soir}\}, u(\text{matin}) = u(\text{soir}) = 0)$, pour tout $i \in \{7, 8\}$
- Les agents travaillant le jour travaillent entre 7 et 10 jours par quinzaine ⇔ les agents sont au repos hebdomadaire entre 4 et 7 jours par quinzaine :
  $c_{fe}(e_i, 1, 14, 14, \{\text{repos}\}, l(\text{repos}) = 4, u(\text{repos}) = 10)$, pour tout $i \in \{1, \ldots, 6\}$
- Les agents travaillant la nuit travaillent entre 5 et 7 jours par 15-aines les agents travaillant la nuit sont au repos hebdomadaire entre 7 et 9 jours par quinzaine :
  $c_{fe}(e_i, 1, 14, 14, \{\text{repos}\}, l(\text{repos}) = 7, u(\text{repos}) = 9)$, pour tout $i \in \{7, 8\}$
- Un agent travaillant l'après-midi d'un jour j, ne travaille pas le matin du jour $j + 1$:
  $c_{ce}(e_i, 1, 14, js, \text{succ}(js), (EV \times EV) \setminus \{(\text{soir, matin})\})$,
  pour tout $i \in \{1, \ldots, 6\}$ et $js \in \{\text{dimanche}, \ldots, \text{samedi}\}$.
  L'opérateur unaire succ définit l'ordre circulaire usuel sur les jours de semaine.
- Un agent ne peut pas travailler plus de 7 jours successifs. S'il travaille 7 jours successifs, il s'ensuit une période de 2 jours de repos minimum :
  $c_{se}(e_i, 1, 14, \{\text{matin, soir}\}, \{\text{repos}\}, 7, 2)$, pour tout $i \in \{1, \ldots, 6\}$.
- Un agent ne peut faire plus de 4 nuits successives. S'il travaille 4 nuits successives, il lui faut au moins 2 jours de repos :
  $c_{se}(e_i, 1, 14, \{\text{nuit}\}, \{\text{repos}\}, 4, 2)$, pour tout $i \in \{7, 8\}$.

L'instance PEPTAG12-84 vérifie en plus les contraintes suivantes :

- Un agent ne peut pas travailler plus de 6 jours successifs.
- Les enchaînements suivants sont interdits : {(soir, matin), (nuit, matin)} sur tout l'horizon.
- Un nombre maximum de 10 semaines consécutives sans repos le dimanche.

8.4.2. Influence des Choix de la Variable et de la Valeur sur la Résolution

La résolution a été effectuée sur un PC Pentium II / 233 MHz / WINDOWS NT 4, 64 Ko RAM. Le choix lexicographique (L) utilisé pour l'instance PEPTAG4-14 est « repos, J8, J9, Jl, Js », alors que pour les deux autres instances, ce choix est « matin, soir, nuit, repos ». Le tableau suivant résume les temps de réponse obtenus pour les différentes stratégies de choix implémentées. Chaque ligne est une instance et chaque colonne est un couple de stratégies de choix (variable, valeur) :

	JL		AL		A		PPDJL		PPDA	
instance	L	A*	L	A*	L*	A*	L	A*	L*	A*
PEPTAG4-14	< 1	< 1	< 1	< 1	< 1	< 1	< 1	< 1	< 1	< 1
PEPTAG8-14	> 3600	5	> 3600	68	> 3600	4	< 1	< 1	< 1	< 1
PEPTAG12-84	> 3600	> 3600	> 3600	> 3600	> 3600	> 3600	> 3600	75	> 3600	74

Tableau 11 : Durée (en secondes) pour trouver une 1ère solution réalisable.
*** = Moyenne sur 20 tests. Stratégie d'évaluation : RFLA.**

Le temps de calcul d'une solution réalisable est plus petit pour les stratégies de choix de variable basées sur le plus petit domaine. Ces résultats confirment ainsi le *first fail principle*[189] qui consiste à choisir la variable qui a le plus de chances de conduire à un échec pour minimiser les coûts de retour arrière.

[189] Pour plus de détails, voir par exemple FROST [1997] dans le cadre des CSP à contraintes binaires.

De même, la stratégie du choix aléatoire A des valeurs de domaines donne significativement de meilleurs résultats de performances que la stratégie lexicographique L.
Parmi les 3 niveaux de consistance (§ 8.3.2.1) le real-full-lookahead est le plus efficace sur nos instances.

8.4.3. Recherche de Solutions Maximales

Nous nous intéressons ici à la durée d'une recherche de solution maximale. Les tests ont été effectués sur l'instance PEPTAG4-14. Le tableau suivant donne les résultats obtenus lorsque la stratégie d'évaluation utilisée est le real-full-lookahead (RFLA) et les stratégies de choix sont le plus petit domaine aléatoire (PPDA) pour la variable et l'aléatoire (A) pour la valeur.

Durée moyenne pour trouver une 1ère solution efficace (secondes)		2, 55
Durée moyenne pour prouver la maximalité ou l'inconsistance (secondes)		11,2
Durée totale moyenne de résolution (secondes)		13,75
%age de tests dont la solution maximale est la :	1e solution réalisable trouvée	10 %
	2e solution réalisable trouvée	20 %
	3e solution réalisable trouvée	55 %
	4e solution réalisable trouvée	15 %
	*n*e solution réalisable trouvée, avec $n > 4$	0 %

Tableau 12 : Résultats obtenus avec l'instance PEPTAG4-14. Moyenne sur 20 tests, avec la stratégie RFLA+PPDA+A.

Les 3 premières lignes du tableau informent que c'est la preuve de l'optimalité qui consomme le plus de temps calcul (80 % du temps total pour l'instance PEPTAG4-14). Le reste du tableau informe sur la répartition du nombre de solutions réalisables trouvés avant d'obtenir une solution Pareto optimale. La hauteur du préordre formant les préférences collectives semble être 4 car une solution maximale est obtenue en au plus 4 itérations. Plus de la moitié des tests (55 %) trouvent la solution maximale en 3 itérations. Le Tableau 13 nous montre une solution maximale pour le PEPTAG4-14.

	L	M	M	J	V	S	D	L	M	M	J	V	S	D
agent 1	R	J9	J9	J9	J8	Js	R	R	J8	R	J8	J8	R	R
agent 2	Jl	J8	R	R	R	R	R	Jl	J8	J8	J9	J8	Js	R
agent 3	R	J8	J8	J8	J8	Js	R	R	J9	J9	J8	R	R	R
agent 4	Jl	J9	R	J8	J9	R	R	Jl	J9	R	R	J9	Js	R

Tableau 13 : Planning Pareto optimal pour l'instance PEPTAG4-14.

De manière à obtenir plus d'information, des tests ont été effectués sur l'instance PEPTAG8-14. Ainsi, lorsqu'une première solution réalisable est obtenue, la contrainte de Pareto est ajoutée. La durée moyenne de recherche d'une solution réalisable passe alors en moyenne de moins d'une seconde à 740 secondes (12 mn) pour la combinaison de stratégies RFLA+PPDA+A, à partir de 20 tests. Deux effets semblent s'additionner pour expliquer cette détérioration des performances. Tout d'abord, le temps de calcul est généralement inversement proportionnel au nombre de solutions réalisables. De plus, l'algorithme de filtrage utilisé pour les VC-contraintes souples de dénombrement de séquences (§ 8.3.2.3) semble influencer très défavorablement la puissance d'élagage du processus de propagation de contraintes. Dans les mêmes conditions, le temps moyen cumulé pour trouver une troisième solution réalisable est de 1344 secondes. Par conséquent, le programme prend en moyenne autant de temps pour calculer la 2e solution réalisable et la 3e. Ce qui crédibilise plus la thèse d'un algorithme de filtrage de mauvaise qualité.

8.4.4. Discussion

Ces résultats appellent à effectuer des tests plus conséquents.

Les stratégies de choix de variables et de valeurs influencent énormément les performances des algorithmes. Il est possible de tirer partie des informations propres aux PEPTAG comme PARTOUCHE [1998, chap. 7] et CHAN [2002, § 3.5] l'ont fait, pour atteindre ce but.

Les algorithmes de recherche peuvent être considérablement améliorés en introduisant par exemple des (méta-)heuristiques d'amélioration locales de la solution comme 2-Opt, recuit simulé (JASZKIEWICZ [2001]), etc. ; ou en utilisant des retours arrières intelligents.

Ces résultats sont obtenus dans un contexte préférentiel très limité. Des travaux futurs seraient intéressants pour connaître leur comportement face à des modèles plus complexes.

Concernant la dominance de Pareto P et celle de Lorenz généralisée GL, par définition, GL est une extension de P. Ce qui se traduit en termes de solutions réalisables par : $S(c_{\mathrm{P}}[\mathsf{mieux}, x], V, D) \subseteq S(c_{\mathrm{GL}}[\mathsf{mieux}, x], V, D)$, avec x une solution (§ 5.3.1.2), les contraintes étant appliquées sur la même VC-contrainte individuelle. Par conséquent, pour un même niveau de consistance la contrainte de Pareto a un pouvoir de réduction de domaines plus fort que la contrainte de Lorenz généralisée. Sachant que les algorithmes de la PPC donnent généralement de meilleures performances lorsque les contraintes ont un fort pouvoir d'élagage (CASEAU *et al*. [1995], PARTOUCHE [1998, § 7.3.2]), on peut supposer que les temps de calcul seront plus grands avec la contrainte GL. A contrario, sachant que les algorithmes de la PPC donnent généralement de meilleurs performances lorsque le nombre de solutions réalisables est grand, on peut alors supposer que la résolution sera plus rapide avec la contrainte GL. Tous les scénarii sont donc possibles. Seuls des tests pourront permettre de trancher.

Pour améliorer globalement les temps de calcul, une possibilité est de concevoir des stratégies d'évaluation orientées vers les solutions de meilleur compromis[190].

Bien d'autres tests et implémentations restent à faire pour répondre à toutes ces questions et pour confirmer ou infirmer les intuitions. D'autres perspectives sont discutées en conclusion.

[190] Voir aussi la fin du § 7.1.3.

Conclusions et Perspectives

Notre objectif principal était de créer un système informatique qui privilégie le dialogue avec les décideurs et qui leur propose de choisir un planning parmi les plus équitables et les plus efficaces. Force est de constater que la partie était loin d'être gagnée. Un certain nombre d'étapes essentielles ont été franchies. Nous les récapitulons ci-dessous :

Récapitulatif.

Tout d'abord, nous avons décidé de revoir fondamentalement les modèles préférentiels sur lesquels sont actuellement bâtis les problèmes d'élaboration de plannings de personnel. La fonction objective en est le composant principal. Or, elle ne modélise que très imparfaitement les préférences humaines, et se révèle être un outil particulièrement inadapté à la représentation de concepts comme l'équité. Nous avons donc proposé une nouvelle représentation des préférences basée sur la relation binaire de préférence (rbp). Par rapport à la fonction objective, la rbp admet l'incomparabilité et l'incertitude, allant de pair avec une description des préférences humaines plus adéquate et une résolution interactive.

Le choix à partir d'une rbp n'est pas toujours possible en appliquant les axiomes de choix usuels : l'optimalité et la maximalité. Nous les avons généralisés au chapitre 3, car les alternatives indiquées dans la littérature sont souvent controversées. Nous avons aussi défini un axiome de redondance préférentielle pour filtrer intelligemment l'information communiquée aux utilisateurs. Des théorèmes de caractérisation ont ensuite été élaborés pour ces nouveaux axiomes, formant ainsi des résultats inédits en théorie des graphes.

En termes ensemblistes, la relation binaire de préférence est un sous-ensemble de couples de solutions. Si les solutions sont données en extension, la rbp peut être représentée par sa liste de couples. Mais lorsque les solutions sont implicites, il est nécessaire d'adopter une représentation implicite de la rbp. Jusqu'à maintenant, seuls des cas particuliers de relations ont été modélisés implicitement. Nous avons décidé de combler ce manque lorsque les solutions sont décrites par un système de contraintes (chapitre 5). La représentation implicite de la rbp définie est appelée *PB-contrainte souple*. Nous avons également établi une définition récursive pour concevoir non seulement des règles d'agrégation (critères de compromis), ces dernières étant les outils privilégiés pour la synthèse d'information préférentielle individuelle en préférences collectives (chapitre 4), mais aussi des hiérarchies agrégatives, lorsque la complexité du modèle préférentiel nécessite plusieurs niveaux d'agrégation.

Nous avons opté pour une démarche de conseil : Aider le décideur à fonder sa décision sur des valeurs objectives universelles telles que l'équité et l'efficacité pour évaluer les plannings dans les problèmes d'élaboration (chapitre 6). Ces concepts s'obtiennent à partir des préférences individuelles des différents acteurs. L'équité est une valeur souvent recherchée, car fédératrice comme l'efficacité (par exemple, la dominance de Pareto), mais difficile à comprendre et à modéliser. Pour se faire, nous nous sommes servis des investigations axiomatiques effectuées en éthique et en TCS ayant conduit à isoler ce concept. Ces investigations permettent de concevoir des règles d'agrégation (la dominance de Lorenz et ses extensions, la règle d'équité de Suppes-Sen) uniquement basées sur les principes sous-jacents (principes de transfert de Pigou-Dalton et de Suppes). Les concepts d'efficacité et d'équité sont avant tout partiels dans leurs évaluations. Pour être plus concret, une solution est dite équitable pour un problème lorsqu'il n'est pas possible de faire des transferts d'un mieux loti vers un moins

bien loti, et obtenir une autre solution réalisable. Il peut exister plusieurs solutions équitables, incomparables entre elles du point de vue de l'équité. Chacun a ses avantages et ses inconvénients non compensables. Dans ce cas, l'appréciation et le choix sont laissés aux décideurs. Laisser le système informatique décider, alors que les solutions sont incomparables, serait priver les acteurs de leur liberté de choisir. D'où l'importance de l'interactivité.

L'équité nécessite qu'il y ait des comparaisons interpersonnelles, conduisant à des modèles préférentiels spécifiques appelés *relation binaire de préférence étendue*. Les PB-contraintes souples ont été adaptées à ce cas de figure.

L'équité et l'efficacité ont été combinées pour définir différentes règles d'agrégation dédiées aux problèmes d'élaboration de plannings de travail, comme la qualité de service aux clients, la qualité de vie des travailleurs.

Dans le cas des programmes mathématiques multiobjectifs, nous avons montré comment utiliser des résultats de la théorie polyédrale pour transformer les problèmes d'allocation de ressources basé sur l'équité évaluant les solutions au moyen de la dominance de Lorenz (et ses extensions), en problèmes évaluant les solutions au moyen de la dominance de Pareto (chapitre 7). Cette transformation, bien qu'elle ne soit pas polynomiale, permet d'utiliser les procédures de résolution interactive développées pour les derniers, en les couplant à des techniques de génération de coupes polyédrales, afin de résoudre les premiers.

L'aspect interactif a aussi été abordé directement, avec les systèmes interactifs d'aide à la décision (SIAD). De nouvelles problématiques d'aide à la décision ont été spécifiées pour les systèmes de contraintes basés sur les préférences (chapitres 3 et 5). Elles couplent le choix de solutions de compromis à la consistance des valeurs de domaines et à l'explication d'incohérence.

Muni de ce nouveau modèle, nous nous sommes appliqués (chapitre 7) à concevoir des algorithmes de recherche de solutions maximales basés sur les procédures de séparation et évaluation (Branch and Bound). Nous avons fondé l'évaluation de nos modèles sur la consistance des valeurs de domaines. Pour se faire, des algorithmes de filtrage ont été créés, dédiés à quatre PB-contraintes souples représentants les préférences collectives de règles d'agrégation équitables et/ou efficaces. Dans ce cadre, un algorithme de filtrage a été conçu pour la dominance de Lorenz, usant de la contrainte de tri. Et, après avoir explicité les relations liant la règle d'équité de Suppes-Sen et les concepts de couplage et de couverture des sommets dans un graphe biparti, nous les exploitons pour confectionner des algorithmes de filtrage pour cette règle.

Pour terminer, certains de ces algorithmes ont été implémentés pour concevoir un prototype de SIAD dédié au problème d'élaboration de plannings infirmiers (chapitre 8). Le modèle préférentiel utilisé est très simple en comparaison de la théorie qui vient d'être développée. Ainsi est-elle formée d'une fonction objective par travailleur, les préférences individuelles étant agrégées par la dominance de Pareto. Trois modes de résolution ont été proposés. Le premier, interactif, maintient un niveau de consistance sur le système de contraintes, les utilisateurs pouvant ajouter ou supprimer des contraintes. Le second recherche une solution réalisable, tandis que le dernier recherche une solution maximale pour la dominance de Pareto. Les performances algorithmiques sur des exemples réels sont présentées.

Perspectives.

Ces travaux constituent un corps de connaissance qui reste embryonnaire, comparé à l'étendue des voies encore inexplorées, qu'elles concernent la modélisation, l'agrégation, la confection d'algorithmes ou la réalisation pratique.

La perspective la plus immédiate serait de construire un prototype complet, et d'effectuer des tests grandeur nature afin d'étudier comment ce nouveau système est perçu par les différents acteurs concernés. Ces tests permettraient de constater si la puissance d'évaluation de l'équité qu'impose le système informatique se marie bien avec les systèmes de valeurs des décideurs ; Et en cas de désaccord, de voir si la modification de l'espace d'évaluation de l'équité est une procédure robuste d'amélioration de l'évaluation de l'équité. En ce qui concerne la complexité des modèles préférentiels, il serait intéressant de connaître son niveau d'influence sur l'efficacité algorithmique.

Par prototype complet, nous entendons aussi le développement d'interfaces pour l'acquisition des préférences implicites. Cette tâche plus qu'auparavant, nécessitera que l'on s'y attarde, elle conditionne le reste du processus. La précision obtenue en utilisant les rbp à la place des fonctions objectives ne devra pas être perdue par des méthodes d'acquisition et de communication inadéquates.

Les résultats des premiers tests effectués nous autorisent à être très optimistes sur les développements algorithmiques futurs. Certes, les problèmes standards de satisfaction de contraintes se résolvent avec une très grande efficacité par rapport aux problèmes de choix basés sur les préférences. Mais, nous mettrons ce manque de performance sur la jeunesse des dernières. Bien d'autres méthodes restent à adapter aux préférences notamment les méta-heuristiques qui couplées aux méthodes de résolution exactes améliorent très significativement les performances algorithmiques (algorithmes hybrides).

Il serait aussi très utile de spécialiser les algorithmes présentés (stratégies de recherche de solutions) aux problèmes d'élaboration de plannings, et d'améliorer les algorithmes de consistance pour les différentes contraintes souples proposées.

Le prototype conçu est limité à l'élaboration de plannings infirmiers. Il serait enrichissant d'ouvrir le spectre des problèmes étudiés dans la pratique comme la planification journalière, annuelle, les problèmes usant de multi-compétence ou pour lesquels le nombre d'employés est inconnu ; et plus généralement la planification sanitaire (PELLETIER [1999]), la gestrion de projet, etc.

De plus, de manière à traiter des problèmes d'équité et d'efficacité plus importants avec la programmation par contraintes, il est nécessaire de développer des algorithmes de filtrage pour les versions multidimensionnelles de la dominance de Lorenz.

Maintenant à long terme et de manière plus générale :

Une autre voie de recherche en théorie des graphes a été ouverte avec les nouveaux axiomes de choix présentés, généralisant l'optimalité et la maximalité. Elles méritent d'être poursuivies. Elles donneront des bases mathématiques plus étendues aux spécialistes de la décision et du choix social pour définir de nouvelles règles d'agrégation et procédures interactives.

De plus en plus d'applications admettent un modèle préférentiel possédant des propriétés structurales très faibles. Pour pouvoir les traiter, il faudrait concevoir des algorithmes pour résoudre des problèmes de choix sur les systèmes de contraintes basés sur les préférences, admettant des PB-contraintes souples ayant des propriétés structurales plus faibles que la transitivité. Cela permettrait de tester entre autre les limites de ce modèle.

Parallèlement au développement d'algorithmes de filtrage pour contraintes dures, il serait intéressant d'en produire aussi pour différentes PB-contraintes souples représentant la rbp

collective de règles d'agrégation. Cela permettrait de diversifier les modes d'agrégation des préférences individuelles.

Bien que les PB-contraintes souples aient été utilisées pour représenter les relations binaires de préférence dans un contexte implicite, leur utilisation peut être élargie à des modèles plus complexes, comme les relations binaires floues, valuées, etc.. Toute cette théorie mérite d'être développée pour réduire les différences entre les modèles utilisables dans un contexte où les solutions sont explicitement données et celui où les solutions sont implicites.

Il serait aussi intéressant de développer des algorithmes pour les nouvelles problématiques d'aide à la décision définies. Elles permettront au décideur d'améliorer les réponses aux questions qu'il se pose au cours du processus. Par exemple, obtenir des réponses même en cas d'incohérence (problèmes sur-contraints), baser sa recherche de solution sur des processus de décision totalement inédits, mettant en scène des raisonnements humains automatiques nouveaux.

Ce manuscrit a contribué à dénoncer l'hégémonie des fonctions objectives en recherche opérationnelle. La description d'un système réel ne nécessite aucunement l'utilisation des fonctions objectives pour représenter les préférences. Certes elles possèdent au moins deux avantages par rapport à la rbp. Le premier concerne le pouvoir de représenter des concepts complexes – comme l'équité – par un nombre unique, alors que la rbp demande à être plus créatif pour la communication des concepts sous-jacents. Le second avantage porte sur l'existence d'algorithmes très performants manipulant les fonctions objectives dans un contexte implicite, héritage de leur ancienneté. Ces algorithmes peuvent être néanmoins utilisés pour les rbp, après une étude axiomatique des modalités de transformation, comme cela est fait en TCS via les théorèmes de caractérisation. Ces deux inconvénients qu'ont les rbp par rapport aux fonctions objectives peuvent être grandement palliés en développant d'une part des interfaces ergonomiques intelligents, et d'autres part de nouveaux algorithmes de traitements.

Pour marquer la différence d'avec les problèmes basés sur un critère analytique unique, certains spécialistes parlent de recherche opérationnelle moderne (voir Équipe MUSTIL [1996, p. 3]), ou d'optimisation mathématique (et combinatoire) ordinale (TSOUKIÀS [1999]).

Ce document montre que des principes éthiques comme l'équité ont toute leur place dans les problèmes de gestion des hommes. Il permet aussi d'établir un dialogue entre différentes communautés, chacun étudiant un aspect différent mais complémentaire des problèmes de décision. Nous espérons que ce travail aura permis de réconcilier le chercheur opérationnel et les scientifiques sociaux et humains. Le cloisonnement des disciplines nous empêche d'accéder à la riche – et au combien passionnante ! – information venant des sciences humaines et sociales. Nous avons énormément à apprendre de cette communauté. Et vice versa. La « société de l'information » nous impose de ne plus nous limiter à notre discipline pour comprendre et apporter des solutions aux problèmes auxquels sont confrontées nos sociétés. Un apport mutuel (complémentarité) entre les différentes sciences, permettra de définir des modèles qui ne se limitent plus à la discipline (disciplines cloisonnées). Cet échange interdisciplinaire est une solution pour obtenir de meilleurs résultats dans l'élaboration d'outils informatiques adaptés.

Un petit pas a été franchi ici, en favorisant un rapprochement des sciences du traitement de l'information, des sciences mathématiques et des sciences humaines et sociales[191]. Ceci pour

[191] Parmi les auteurs référencés dans ce document, trois sont des lauréats du prix Nobel d'économie : Kenneth ARROW (en 1972), Herbert A. SIMON (en 1978) et Amartya K. SEN (en 1998). Chacun est caractérisé par son travail original : ARROW a introduit les relations binaires de préférences dans les sciences économiques des an-

confectionner des outils informatiques plus pragmatiques, ayant des bases mathématiques solides, et éthiquement plus viables. De tels outils encourageront la tolérance et accorderont aux décideurs des concepts forts pour appuyer leur décision et leur donner l'occasion d'être un peu mieux au service des autres.

---

nées 50 qui étaient exclusivement basées sur les fonctions objectives (créant par la même occasion la TCS). SIMON, l'un des pères fondateurs de l'intelligence artificielle, est l'un des premiers à étudier la place de l'ordinateur dans la résolution des problèmes de décision dans les organisations. Et SEN cet économiste philosophe, a choisi d'introduire plus d'humanité dans les sciences économiques en définissant l'éthique comme denrée essentielle de la décision économique.

Annexes

1. Annexe A1 : Description de Quelques Contraintes Globales de la Littérature

- Contrainte Globale de Différence.

La **contrainte globale de différence** $c_{\text{ALLDIFF}}(V, D)$, dans sa définition la plus générale, considère en entrée :

$V = \{v_1, \ldots, v_n\}$ un ensemble de variables non vide en extension, et

D un ensemble de domaines, associant à chaque variable v de V, un domaine $D(v)$ donné en extension ou non.

Les solutions réalisables correspondent aux éléments de $\mathcal{D}_V = D(v_1) \times \ldots \times D(v_n)$ où les composantes sont deux à deux différentes :

$$S(c_{\text{ALLDIFF}}(V, D)) = \{x \in \mathcal{D}_V \text{ tel que } x(i) \neq x(j) \text{ pour tout } 0 < i < j \leq n\}.$$

RÉGIN [1994] est le premier à avoir défini la contrainte dans le cas des domaines en extension, et proposé un algorithme résolvant le problème d'inverse consistance globale associée. Ensuite, PUGET [1998] s'est intéressé au cas des intervalles fermés sur un ensemble discret totalement ordonné en présentant un algorithme résolvant le problème de borne-consistance globale.

- GCC.

La **contrainte de cardinalité globale** $c_{\text{GCC}}(V, D, B, [l, u])$ considère en entrée :

$V = \{v_1, \ldots, v_n\}$ un ensemble non vide de variables en extension ;

B un ensemble fini non vide en extension ;

D un ensemble de domaines, associant à chaque variable v de V un domaine en extension $D(v) \subseteq B$; et

$[l, u]$ un pavé de $^{\text{card}(B)}$.

Les solutions réalisables sont les éléments de $\mathcal{D}_V = D(v_1) \times \ldots \times D(v_n)$ comportant pour toute valeur d dans B un nombre de composantes affectées à cette valeur compris dans l'intervalle $[l(d), u(d)]$:

$$S(c_{\text{GCC}}(V, D, B, [l, u])) = \left\{ \begin{array}{l} x \in \mathcal{D}_V \text{ tel que : pour tout } d \in B, \\ \qquad l(d) \leq \text{card}\left(\{v \in V \text{ tel que : } x(v) = d\}\right) \leq u(d) \end{array} \right\}$$

RÉGIN [1996] a défini cette contrainte et proposé un algorithme résolvant le problème d'inverse consistance globale associée à celle-ci.

2. Annexe A2 : Rappel sur les Niveaux de Consistance Inverse.

2.1. Préliminaires.

Au § 3.2.3.4 la problématique technique de la consistance a été définie, ainsi que deux problèmes d'inverse consistance : la borne-consistance et l'inverse-consistance globale (GIC). Rappelons qu'une procédure résolvant le problème GIC consiste à supprimer toute valeur de domaine ne faisant pas partie d'une solution réalisable d'un système de contraintes. Le système de contraintes obtenu possède le même ensemble de solutions réalisables que le système initial. Dans la pratique, il est souvent très difficile d'établir un algorithme performant établissant la GIC sur un système de contraintes. Dans le cadre des systèmes de contraintes formés de domaines en extension, des algorithmes très généraux pour la GIC ont été proposés (FREUDER [1985], COOPER [1989] VERFAILLIE *et al*. [1999]). Ces algorithmes ont cependant une complexité théorique et pratique qui les rend inutilisables pour résoudre des problèmes réels. Dans la pratique, un ensemble de problèmes de consistance inverse très variés supprimant une partie des valeurs ne conduisant à aucune solution réalisable ont été introduits : l'arc-consistance, le forward-checking, le real-full-lookahead, la *k*-inverse-consistance, etc..

Historiquement, ces problèmes effectuant une suppression partielle de valeurs *inconsistantes* par rapport à GIC, ont été définis avant le problème GIC.

Au § A2.2 nous présentons différents problèmes de consistance inverse apparus en PPC et basés sur le problème d'appartenance associé au CSP. Ensuite (§ A2.3), les problèmes de consistance inverse de la PPC basés sur les contraintes globales sont exposés. Ceux conçus en MIP sont introduits au § A2.4.

2.1.1. Quelques Définitions.

Donnons quelques définitions. Considérons un système de contraintes $\mathcal{R} = (V, D, C)$.
Une **valeur** d pour la variable v ($\Leftrightarrow d \in D(v)$) est **consistante avec une contrainte** $c \in C$ si et seulement si :

- la variable $v \notin V(c)$, ou
- il existe une solution réalisable x_c pour la contrainte c ($\Leftrightarrow x_c \in S(c)$) telle que $x_c(v) = d$.

Une **variable** v est **consistante avec une contrainte** $c \in C$ si et seulement si :

- le domaine de la variable est non vide : $D(v) \neq \emptyset$, et
- toutes les valeurs possibles pour la variable sont consistantes avec c.

Une **contrainte** c satisfait la propriété d'**inverse consistance globale** si et seulement si toute variable $v \in V(c)$ est consistante avec c. Nous dirons plus simplement que c est GIC. Le problème GIC sur le système de contraintes $\mathcal{R}(c) = (V(c), D(V(c)), \{c\})$ est appelé problème d'inverse consistance globale associée à la contrainte c. Pour simplifier l'écriture, nous écrivons que l'instance de GIC est (c, V, D), ou $c(\gamma)$ lorsque la contrainte est globale, qu'elle reçoit en paramètres γ et qu'il n'y a pas d'ambiguïté sur les domaines. De plus, nous écrivons que GIC retourne l'ensemble de domaines D_{GIC} au lieu du système de contraintes $(V(c), D_{\text{GIC}}, \{c\})$.

2.2. Consistance Inverse et Problème d'Appartenance

La consistance inverse s'appuie sur le principe consistant à dire que, moins il y a de valeurs ne conduisant à aucune solution réalisable, plus un système de contraintes est facile à résoudre via un processus de décision. Ce processus peut être partiellement ou entièrement automatisé, et est basé sur la restriction du domaine réalisable. Comme exemple de processus, nous avons les méthodes arborescentes.

Présentons maintenant l'arc-consistance :

2.2.1. L'Arc-Consistance

Un système de contraintes est **arc consistant** si et seulement si toutes ses contraintes sont GIC.

L'arc-consistance n'est pas équivalent à l'inverse consistance globale. La Figure 55 donne un exemple de système de contraintes arc-consistant, mais pas globalement inverse consistant. En effet, $S_{CSP}(\mathcal{R}) = \emptyset$,alors que toutes les contraintes sont globalement inverse consistantes ($R(\{v_i \neq v_j\}) = \{(a, b), (b, a)\}$ pour tout $i, j \in \{1, 2, 3\}$ et $i \neq j$). Par conséquent, l'arc-consistance n'implique pas l'existence de solution réalisable pour le CSP, elle ne fait que supprimer certaines valeurs ne conduisant pas à une solution réalisable. Certaines valeurs et combinaisons de valeurs contenues dans aucune solution subsistent. Par contre l'inverse consistance globale pour un système implique l'arc-consistance.

Système de contraintes $\mathcal{R} = (V, D, C)$ avec :
$V = \{v_1, v_2, v_3\}$
$D = \{D(v_i) = \{a, b\}$, pour $i = 1, 2, 3\}$
$C = \{v_1 \neq v_2 ; v_1 \neq v_3 ; v_2 \neq v_3\}$

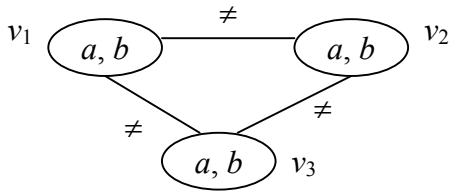


Figure 55 : Exemple de système de contraintes arc-consistant, mais pas globalement inverse consistant.

Le problème qui, à partir d'un système de contraintes quelconque, définit un système de contraintes équivalent et arc-consistant, si un tel système existe, est appelé **problème d'arc-consistance** (AC) :

AC : Étant donné un système de contraintes $\mathcal{R} = (V, D, C)$, retourner le système de contraintes $\mathcal{R}_{AC} = (V, D_{AC}, C)$ arc-consistant équivalent ($\Leftrightarrow S_{CSP}(\mathcal{R}_{AC}) = S_{CSP}(\mathcal{R})$) s'il existe, sinon retourner "non".

Remarquons que $\mathcal{R}_{AC}$ est unique.

Dans le cadre des systèmes de contraintes à domaines finis en extension, il existe différents algorithmes génériques permettant de résoudre le problème d'arc-consistance. La plupart des algorithmes supposent que seul le problème d'appartenance (§ 2.1.1.1) est disponible pour chaque contrainte. Ce problème est supposé polynomial pour toute contrainte[192], et l'algorithme utilisé est compté comme une opération élémentaire. La notation mathématique est directement utilisée : « $x_c \in S(c)$ » pour exprimer que l'algorithme d'appartenance retourne "oui", et « $x_c \notin S(c)$ » dans le cas contraire.

L'algorithme d'arc-consistance le plus utilisé est basé sur une technique incrémentale appelée **propagation de contraintes**. Son principe est le suivant :

[192] Dire que le problème d'appartenance associé à un problème Π est polynomial, est équivalent à dire que Π est une fonction problème associée à NP. En d'autres termes, il existe une Turing réduction de Π vers le problème de décision associé, ce dernier étant dans NP. Voir sur ce point PAPADIMITRIOU [1994, § 10.3].

1) Pour chaque contrainte, réviser le domaine des variables impliquées par la contrainte ; c'est-à-dire supprimer les valeurs du domaine qui ne sont pas consistant avec la contrainte.
2) propager l'étape 1) à toutes les contraintes impliquant les variables pour lesquelles le domaine a été réduit.

Le processus se répète jusqu'à ce que plus aucune réduction n'ait lieu, qu'aucune contrainte ne soit à reconsidérer.

Ces deux étapes se traduisent par les deux algorithmes suivants : `révise` correspond à l'étape 1), et `GAC3` à l'étape 2).

`révise`(v_i : Variable, $c = (V(c) = \{v_1, ..., v_i, ..., v_r\}, r = \text{card}(V(c)), S(c))$: Contrainte) : Booléen
début
  b : Booléen ← non booléen qui vaut oui si au moins une valeur du domaine de x_i a été supprimée.
  pour tout ($d \in D(v_i)$) **faire**
    si ($\forall\ (d_1, ..., d_r) \in D(v_1) \times ... \times D(v_{i-1}) \times D(v_{i+1}) \times ... \times D(v_r)$ on a $(d_1, ..., d, ..., d_r) \notin S(c)$) **alors**
      $D(v_i) \leftarrow D(v_i) \setminus \{d\}$
      $b \leftarrow$ oui
    fin si
  fin pour
  retourner b
fin `révise`

`GAC3`($\mathcal{R} = (V, D, C)$: système de contraintes) : {système de contraintes, non}
début
  Q : ensemble de contraintes ← C Contient tout au long de l'algorithme les contraintes à propager, initialisée à l'ensemble C des contraintes du système $\mathcal{R}$.
  tant que ($Q \neq \emptyset$) **faire**
    Sélectionner une contrainte c de Q
    pour tout ($v \in V(c)$) **faire**
      si (`révise`(v, c) = oui) **alors**
        si ($D(v) = \emptyset$) **alors** **retourner** non **fin si**
        $Q \leftarrow Q \cup \{ c'$ telles que : $v \in V(c') \}$
        $Q \leftarrow Q \setminus \{c\}$
      fin si
    fin pour
  fin tant que
  retourner $\mathcal{R}$ la propagation s'est bien passée.
fin `GAC3`

Algorithme 12 : Établissement de l'arc-consistance sur un système de contraintes, en utilisant un algorithme d'appartenance pour chaque contrainte.

L'algorithme `GAC3` est une généralisation aux contraintes non-binaires de l'algorithme de propagation `AC-3` proposé par MACKWORTH [1977]. Sa complexité en temps dans le plus mauvais cas est $O(mr^2d^{r+1})$, si on note m le nombre de contraintes dans le système, r l'arité maximale des contraintes, et d la taille du domaine le plus grand.

Plusieurs algorithmes procédant par propagation ont été proposés. Citons :

– AC-4 (MOHR & HENDERSON [1986]) pour les contraintes binaires, généralisé au cas non-binaire dans MOHR & MASINI [1988] avec l'algorithme GAC4. GAC4 implique une complexité temporelle dans le plus mauvais cas de $O(md^r)$, mais possède un inconvénient : Il utilise des structures de données stockant des informations intermédiaires, montant sa complexité spatiale à $O(mrd^r)$; ce qui le rend inutilisable pour de grands systèmes de contraintes (BESSIÈRE & RÉGIN [1997]).
– AC-7 (BESSIÈRE *et al.* [1995]) pour les contraintes binaires, généralisé au cas non-binaire dans BESSIÈRE & RÉGIN [1997] avec l'algorithme GAC-schema. GAC-schema a une complexité temporelle dans le plus mauvais cas de $O(md^r)$, et spatiale de $O(mr^2d)$.

D'autres niveaux de consistance ont été proposés intermédiaires entre l'arc-consistance et la consistance inverse globale, comme la *k*-inverse consistance. Pour plus d'informations, nous renvoyons à FREUDER [1985] et VERFAILLIE et *al.* [1999]. Dans la suite, nous nous interressons à des niveaux d'inverse consistance plus faibles que l'arc-consistance.

2.2.2. Real-Full-Lookahead et Forward-Checking

Ce sont des niveaux de consistance inférieurs à l'arc-consistance. Elles sont dites locales car elles dépendent d'un ensemble de variables $V_1 \subseteq V$, et propagent les contraintes se trouvant dans le voisinage de V_1. Selon la définition de cette notion de voisinage, différents niveaux de consistance peuvent être définis :

– Le **forward-checking**[193] (FC) classique déclenche une seule fois les contraintes c dont l'ensemble des variables associé $V(c)$ a une intersection non vide avec V_1 ;
– Le **real-full-lookahead** (RFLA) déclenche les contraintes du forward-checking classique, les contraintes des variables modifiées par l'élagage des premières contraintes et ce itérativement, jusqu'à ce qu'aucune contrainte déclenchée ne modifie le domaine d'une variable.

Pour d'autres niveaux de consistance du type FC, nous renvoyons à BESSIÈRE *et al.* [1999]. Ces niveaux de consistance sont particulièrement bien adaptés aux problèmes de maintien de consistance, car elles peuvent être utilisées pour déclencher uniquement les contraintes se trouvant dans le voisinage des variables modifiées.

Le RFLA possède une propriété qui le lie à l'arc-consistance : Si le système de contraintes initial est arc-consistant, sauf pour les variables de V_1 alors l'établissement du RFLA sur ce système de contraintes en V_1 fournit un système de contraintes arc-consistant. Par conséquent, dans une procédure itérative utilisant le maintien de l'arc-consistance, il suffit d'appliquer cet algorithme sur le système initial, et à chaque itération d'appliquer le RFLA sur les variables qui ont été modifiées par l'ajout d'une contrainte.

Remarquons que AC = RFLA(V). Plus précisément, si l'on note $C_{RFLA(V1)}$ l'ensemble des contraintes de C déclenché par le RFLA en V_1, alors l'égalité $C_{RFLA(V1)} = C$ implique que AC = RFLA(V_1).

Cette propriété a été utilisé au chapitre 8 pour établir l'arc-consistance pour les problèmes de planification infirmière.

2.3. Inverse-Consistance et Contraintes Globales

De même qu'il existe plusieurs niveaux de consistance pour les systèmes de contraintes, il en existe pour les contraintes. Au lieu d'utiliser des algorithmes d'appartenance pour chaque

[193] Le forward-checking et le real-full-lookahead désignent usuellement le couple formé de la procédure de backtrack et de l'algorithme de consistance. Nous les utiliserons pour désigner uniquement le problème de consistance résolu.

contrainte afin d'établir un niveau de consistance sur un système de contraintes, des algorithmes de filtrage peuvent être employés. Ainsi, des algorithmes résolvant des problèmes d'inverse consistance pour chaque contrainte, ont été conçus. Ils sont connus sous le nom de contraintes globales. Voici quelques niveaux de consistance usuels pour une contrainte c :

- le niveau de consistance le plus bas obtenu comme suit : Lorsque les variables de $V(c)$ sont instanciées, tester l'appartenance de la solution identifiée par le domaine réalisable de c : $\mathcal{D}_{V(c)} = \{x\} \subseteq S(c)$?
- le niveau établit en testant uniquement si les variables instanciées forment une solution réalisable partielle pour la contrainte,
- la globale inverse consistance, etc.

Ainsi, au lieu d'utiliser la contrainte uniquement comme vérificateur de solution (algorithme de résolution de problème d'appartenance), celle-ci peut être dotée d'une méthode de filtrage de domaines, afin d'être utilisée comme filtre (ou algorithme d'élagage). Cette méthode, appelée `filtrer`, a pour but de supprimer les valeurs inconsistantes des domaines des variables à sa portée.

Supposons donc que pour chaque contrainte c, nous possédons un algorithme permettant de résoudre un problème d'inverse consistance, que nous notons κ-IC(c), au lieu du traditionnel problème d'appartenance. L'algorithme `GAC3` peut être modifié de manière à utiliser uniquement des algorithmes de filtrage. L'algorithme obtenu est le suivant :

`GAC3_filtre`($\mathcal{R}$= (V, D, C) : système de contraintes) : {système de contraintes, non}
début
  Q : ensemble de contraintes ← C Contient tout au long de l'algorithme les contraintes à propager, initialisée à l'ensemble C des contraintes du système $\mathcal{R}$.
  P : ensemble de variables ← ∅ variables dont le domaine a été modifié pendant le filtrage courant ; initialisée à la liste vide.
  tant que ($Q \neq \emptyset$) **faire**
    Sélectionner une contrainte c de Q
    si (`filtrer`(c, P) = non) **alors**
      retourner non La contrainte c n'est pas consistante.
    sinon
      $Q \leftarrow Q \bigcup_{v \in P} \{c' \text{ telles que } v \in V(c')\}$
      $Q \leftarrow Q \setminus \{c\}$
      $P \leftarrow \emptyset$
    fin si
  fin tant que
  retourner $\mathcal{R}$ La propagation s'est bien passée.
fin `GAC3_filtre`

Algorithme 13 : Établissement d'un niveau d'inverse consistance sur un système de contraintes, en utilisant un algorithme de filtrage pour chaque contrainte.

Cet algorithme utilise la procedure `filtrer` qui reçoit en paramètres la contrainte c et un ensemble de variables P qui sert uniquement à stocker les variables de $V(c)$ qui sont modifiées pour établir le niveau de consistance souhaité.

Cet algorithme a été utilisé dans Joseph [1998] pour résoudre un problème d'élaboration de plannings infirmiers (voir aussi chap. 8). BESSIÈRE & RÉGIN [1999] ont proposé une version filtre de `GAC-schema`.

Remarquons que l'algorithme `GAC3_filtre` établit l'arc consistance sur un système de contraintes si et seulement si le niveau d'inverse consistance κ-IC(c) de chaque contrainte c est égal à GIC. Dans le cas contraire, le niveau de consistance établi sur le système est plus

faible. Par exemple, si notre système est formé uniquement de contraintes globales de différence ALLDIFF (Annexe A1), alors `GAC3_filtre` utilisé avec l'algorithme de filtrage de Régin [1994] établit l'arc consistance sur notre système.

Toute valeur supprimée par l'algorithme `GAC3_filtre` utilisant un niveau d'inverse consistance arbitraire pour chaque contrainte est aussi supprimée par `GAC3_filtre` utilisant exclusivement le niveau GIC pour chaque contrainte ; ce dernier établissant l'arc consistance sur un système de contraintes.

2.4. Consistance Inverse en Programmation Entière.

Nous avons vu au § 3.2.3.4 que la consistance était quelques fois utilisé en RO dans le cadre de la gestion de projet. Elle est aussi utilisée comme étape préliminaire aux algorithmes de Branch-and-Bound pour réduire l'espace de recherche et fixer des variables (voir SAVELSBERGH [1994]).

SOLNON [1997*a*] et [1977*b*] utilisa la programmation linéaire pour établir un niveau d'inverse consistance, nommé LIB-consistance, sur les systèmes d'inégalités linéaires en nombres mixtes. Son principe consiste à établir un niveau de consistance inverse, par l'intermédiaire de la relaxation entière (= obtenu par suppression des contraintes d'intégrité) du système initial. Ce niveau de consistance est généralement plus faible que la consistance inverse globale GIC.

3. Annexe B1 : Preuves de Propositions du Chapitre 3

Proposition 28 : Étant donné un ensemble non vide S structuré par une relation binaire de préférence $\succcurlyeq$, alors l'assertion suivante est vraie :
($\succcurlyeq$ est transitive) $\Rightarrow$ (A est un noyau $\Leftrightarrow$ A est un stable maximal et respecte la maximalité)

Preuve :

$\Leftarrow$ Tout d'abord, remarquons que A est un stable de $M(S, \succcurlyeq)$. Est-il un stable de S ? Étant donné que $M(S, \succcurlyeq)$ est un sous-graphe de S, alors $\forall\ x, y \in A$, $x \parallel y$ est aussi valable pour S. Par conséquent, A est un stable de S.
Montrons maintenant que A est un ensemble dominant.
En premier lieu, l'ensemble A étant un stable maximal de $M(S, \succcurlyeq)$, alors $\forall\ x \in M(S, \succcurlyeq) \setminus A$ et $\exists\ y \in A$, vérifiant $x \simeq y$.
Ensuite, montrons que $\forall\ z \in S \setminus M(S, \succcurlyeq)$, il existe un élément x de A tel que : $x \succcurlyeq z$. En premier lieu, A étant inclus dans $M(S, \succcurlyeq)$ alors $\forall\ x \in A$, $\mathsf{non}[x \prec z]$.
Maintenant, s'il existait un $z \in S \setminus M(S, \succcurlyeq)$ tel que $\forall\ x \in A$ $\mathsf{non}[x \succcurlyeq z]$, alors $x \parallel z$ (sachant que $x \parallel z \Leftrightarrow \mathsf{non}[x \succcurlyeq z]$ et $\mathsf{non}[x \prec z]$). Comme $z \in S \setminus M(S, \succcurlyeq)$ alors il existe un $y \in M(S, \succcurlyeq) \setminus A$ tel que $z \prec y$. Par conséquent, selon la table de transitivité des préordres partiels (Tableau 14), $[x \parallel z$ et $z \prec y] \Rightarrow x \prec y$ ou $x \parallel y$, $\forall\ x \in A$. Comme x et y sont des solutions maximales alors seul le cas $x \parallel y$ existe, et ce pour tout $x \in A$. Ce qui contredit le fait que A est un stable maximal dans $M(S, \succcurlyeq)$, car $A \cup \{y\}$ est un stable dans $M(S, \succcurlyeq)$. Par conséquent il n'existe pas de $z \in S \setminus M(S, \succcurlyeq)$ tel que $\forall\ x \in A$, $\mathsf{non}[x \succcurlyeq z]$. Ainsi, A est un ensemble dominant.
En conclusion, A est un dominant stable de S, c'est-à-dire un noyau de S.

T_{POP}	$\simeq$	$\succ$	$\prec$	$\parallel$
$\simeq$	$\{\simeq\}$	$\{\succ\}$	$\{\prec\}$	$\{\parallel\}$
$\succ$	$\{\succ\}$	$\{\succ\}$	$AF(RBP)$	$\{\succ, \parallel\}$
$\prec$	$\{\prec\}$	$AF(RBP)$	$\{\prec\}$	$\{\prec, \parallel\}$
$\parallel$	$\{\parallel\}$	$\{\succ, \parallel\}$	$\{\prec, \parallel\}$	$AF(RBP)$

Tableau 14 : Table de transitivité des préordres partiels (BELMANDT [1993]).

$\Rightarrow$ Avant tout, montrons que A est un ensemble satisfaisant la maximalité : Montrons qu'il n'existe aucun couple $x \in A$ et $y \in S \setminus A$ tel que $x \prec y$. En effet, si un tel couple existait, A étant un ensemble dominant, alors $\exists\ z \in A$ tel que $z \succcurlyeq y$ (comme $y \in S \setminus A$). La transitivité de $\succcurlyeq$ (voir table de transitivité Tableau 14) implique alors que $z \succ x$. Ce qui contredit le fait que $z \parallel x$ (x et z sont deux éléments d'un stable de S). Par conséquent, A est un ensemble satisfaisant la maximalité.
De plus, A étant un stable de S, A est aussi un stable de $M(S, \succcurlyeq)$ étant donné qu'aucun arc n'est ajouté.
Enfin, A est stable maximal dans $M(S, \succcurlyeq)$. En effet, s'il existe un $y \in M(S, \succcurlyeq) \setminus A$ tel que $\forall\ z \in A$, $z \parallel y$, alors il n'existe pas de $x \in A$ tel que $x \succcurlyeq y$. Ce qui contredit l'hypothèse que A est un ensemble dominant. Par conséquent, A est un stable maximal dans $M(S, \succcurlyeq)$. ❑

Proposition 2 : Étant donnés deux e.r.o.m.i. différents MRO_1 et MRO_2 d'un ensemble non vide S structuré par une relation binaire de préférence $\succcurlyeq$, alors

(a) Deux solutions qui ne font pas partie du même e.r.o.m.i. sont indifférentes :

$$\forall\ (x, y) \in (MRO_1 \setminus MRO_2) \times (MRO_2 \setminus MRO_1),\quad x \simeq y$$

(b) Leur intersection est vide : $MRO_1 \cap MRO_2 = \varnothing$

Notons MRO, l'un des deux e.r.o.m.i..

(c) Si $\mathrm{card}(MRO) \geq 2$ alors $\forall$ solution $x \in MRO$, $\exists\ y \in MRO$ telle que $\mathrm{non}[x \simeq y]$

(d) Si $\mathrm{card}(MRO) \geq 3$ et que pour une solution $x \in MRO$, $\exists\ y \in MRO$ telle que $x \succ y$ (resp. $x \prec y$), alors $\exists\ z \in MRO \setminus \{x, y\}$ telle que $\mathrm{non}[x \succcurlyeq z]$ (resp. $\mathrm{non}[z \succcurlyeq x]$)

(e) Si S est fini, alors $\forall\ x, y \in MRO$, soit $x = y$, soit $\exists\ x_1, \ldots, x_{r-1} \in MRO$ avec $r \geq 1$, tel que $\mathrm{non}[x \succcurlyeq x_1]$, $\mathrm{non}[x_i \succcurlyeq x_{i+1}]$ pour tout $1 < i < r$, et $x_r = y$.

Preuve : Tout d'abord remarquons que $MRO_1 \setminus MRO_2 \neq \varnothing$ et $MRO_2 \setminus MRO_1 \neq \varnothing$ ($\Leftrightarrow$ l'un ne peut pas être inclus dans l'autre), car MRO_1 et MRO_2 sont différents et sont des e.r.o.m.i. (minimaux par inclusion), donc ne peuvent pas contenir un ensemble relativement optimal. Montrons maintenant chaque proposition.

(a) Pour démontrer que $x \simeq y$, montrons en premier que $x \succcurlyeq y$, puis que $y \succcurlyeq x$. Nous avons $x \succcurlyeq y$ par définition de MRO_1 : $x \in MRO_1$ et $y \in MRO_2 \setminus MRO_1 \subseteq S \setminus MRO_1$. Il en est de même pour $y \succcurlyeq x$ par définition de MRO_2 : $y \in MRO_2$ et $x \in MRO_1 \setminus MRO_2 \subseteq S \setminus MRO_2$. D'où le résultat.

(b) Raisonnons par l'absurde. Supposons qu'$\exists\ x \in MRO_1 \cap MRO_2$. Par conséquent, $x \succcurlyeq y\ \ \forall\ y \in MRO_1 \Delta MRO_2$ (différence symétrique de MRO_1 et MRO_2), par définition de MRO_1 et MRO_2. Ce qui implique $MRO_1 \cap MRO_2$ est un e.r.o.. Or MRO_1 et MRO_2 sont minimaux par inclusion, pourtant ils contiennent strictement un e.r.o. ; ce qui contredit l'hypothèse de départ. Ainsi, $MRO_1 \cap MRO_2 = \varnothing$.

(c) Raisonnons par l'absurde. Supposons qu'un tel y n'existe pas. Donc $x \simeq z$, $\forall\ z \in MRO$. De plus, $x \succcurlyeq z$, $\forall\ z \in S \setminus MRO$. En définitive, $x \succcurlyeq z$, $\forall\ z \in S$; et donc $\{x\}$ est un e.r.o.m.i. inclus strictement dans MRO qui est aussi un e.r.o.m.i.. Ce qui contredit la définition d'un e.r.o.m.i.. Par conséquent, un tel y existe.

(d) Étudions le cas où $x \succ y$. Pour cela, raisonnons par l'absurde. Supposons qu'un tel z n'existe pas. Par conséquent, $\forall\ z \in MRO \setminus \{x, y\}$, $x \succcurlyeq z$. De plus, comme $x \in MRO$, alors $\forall\ z \in S \setminus MRO$, $x \succcurlyeq z$. En définitive, $x \succcurlyeq z$, $\forall\ z \in S$; Et l'on revient à la même contradiction que dans (c) : $\{x\}$ est un e.r.o. inclus strictement dans un e.r.o.m.i. MRO. Par conséquent, un tel z existe.

Étudions le second cas : $x \prec y$. Raisonnons encore par l'absurde. Supposons donc qu'un tel z n'existe pas. Par conséquent, $\forall\ z \in MRO \setminus \{x, y\}$, $z \succcurlyeq x$. Par conséquent, $MRO \setminus \{x\}$ est un e.r.o. strictement inclus dans MRO qui est un e.r.o.m.i.. Ce qui contredit la définition d'un e.r.o.m.i.. Par conséquent, un tel z existe.

(e) Cette assertion est une conséquence directe d'un résultat bien connu de la théorie des graphes, qui est présenté tout de suite :

Tout d'abord, étant donné un graphe orienté $G = (U, A)$, un **cocircuit** est un sous-ensemble d'arcs $\mathcal{C} \subseteq A$ pour lequel on peut trouver $B \subset U$ vérifiant :

$$\mathcal{C} = \{(z, t) \in A \text{ tel que } z \in B \text{ et } t \in U \setminus B\} \quad \text{et} \quad \{(z, t) \in A \text{ tel que } z \in U \setminus B \text{ et } t \in B\} = \varnothing$$

Le théorème suivant permet de relier la notion de chemin, à celle de cocircuit et de forte connexité (§ 2.1.2) dans un graphe orienté. C'est un corollaire du fameux lemme de Minty (voir par exemple SAKAROVITCH [1984*a*, § 5.2]).

Théorème 7 : Pour un graphe G connexe contenant au moins un arc, les conditions suivantes sont équivalentes :

(i) G est fortement connexe
(ii) Tout arc appartient à un circuit
(iii) G ne comporte pas de cocircuit.

Nous pouvons alors démontrer l'assertion (e).

Preuve de l'assertion (e) de la Proposition 2 : Considérons le graphe orienté $(MRO, \succcurlyeq_1)$ obtenu à partir de $(MRO, \succcurlyeq)$ comme suit :

$$z \succcurlyeq_1 t \;\Leftrightarrow\; \text{non}[z \succcurlyeq t] \qquad \text{pour } (z, t) \in MRO. \qquad \text{(B1–53)}$$

En premier lieu, en s'aidant du théorème précédent, remarquons que l'assertion (e) est équivalente à dire que : si MRO est un e.r.o.m.i. alors le graphe $(MRO, \succcurlyeq_1)$ est fortement connexe.

Maintenant, pour effectuer la preuve de (e), raisonnons par l'absurde. Deux cas se présentent : $(MRO, \succcurlyeq_1)$ est connexe et MRO n'est pas connexe. Étudions chaque cas :

– Supposons que $(MRO, \succcurlyeq_1)$ ne soit pas connexe. Alors il existe une bipartition (M_1, M_2) de MRO telle qu'il n'existe aucune arc entre M_1 et M_2. En utilisant la définition (B1–53), Par conséquent, pour tout couple $(z, t) \in M_1 \times M_2$, $z \simeq t$. Ce qui signifie que M_1 et M_2 sont deux e.r.o. de $(S, \succcurlyeq)$ car $\forall\, i \in \{1, 2\}$ et $\forall\, (z, t) \in M_i \times (S \setminus M_i)$, $z \succcurlyeq t$. Mais cette déduction contredit le fait que MRO soit un e.r.o.m.i..
– Supposons que $(MRO, \succcurlyeq_1)$ soit connexe, mais pas fortement connexe. Ce qui est équivalent à dire (Théorème 7) qu'il existe un cocircuit $\mathcal{C}$ dans $(MRO, \succcurlyeq_1)$. Notons $M_1(\mathcal{C})$ et $M_2(\mathcal{C})$ la bipartition de MRO effectuée par le cocircuit $\mathcal{C}$:
$M_1(\mathcal{C}) = \{z \in MRO \text{ tel que : } \exists\, t \in MRO \text{ et } (z, t) \in \mathcal{C}\}$ et $M_2(\mathcal{C}) = MRO \setminus M_1(\mathcal{C})$.
En utilisant la formule (B1–53) le graphe $(MRO, \succcurlyeq_1)$ contient un cocircuit est équivalent à dire que :

$$\text{pour tout couple } (z, t) \in M_1(\mathcal{C}) \times M_2(\mathcal{C}), \quad \left.\begin{array}{c} \text{non}[z \succcurlyeq_1 t] \;\text{ ou} \\ (\text{non}[z \succcurlyeq_1 t] \;\text{ et }\; \text{non}[t \succcurlyeq_1 z]) \end{array}\right\} \Leftrightarrow z \succcurlyeq t.$$

Comme $M_1(\mathcal{C}) \subset MRM$, alors par définition, $M_1(\mathcal{C})$ est un e.r.o. de $(S, \succcurlyeq)$, ce qui contredit le fait que MRO soit un e.r.o.m.i.

En résumé, les deux possibilités conduisent à une contradiction. Par conséquent, si MRO est un e.r.o.m.i. alors l'assertion (e) est vraie. ❑

Proposition 3 : Un e.r.o. MRO est un e.r.o.m.i. ssi (a) il est réduit à un élément, (b) il est formé de deux éléments incomparables, (c) il est de cardinal supérieur à 2 et vérifie les assertions (c) à (e) de la Proposition 2.

Preuve : Pour les assertions (a) et (b) la preuve est immédiate. Concernant l'assertion (c), remarquons que (e) implique (c) et (d) de la Proposition 2. Donc, il suffit de montrer que MRO est un e.r.o.m.i. si et seulement si MRO est un e.r.o. et que vérifiant l'assertion (e) de la Proposition 2 est vérifié. La Proposition 2 a montré l'implication dans le sens direct. La réciproque est, elle, directement obtenue à partir de la démonstration effectuée pour (e), car nous avons utilisé que des équivalences. ❑

Proposition 4 : Étant donné deux e.r.m.m.i. différents MRM_1 et MRM_2 d'un ensemble non vide S structuré par une relation binaire de préférence $\succcurlyeq$, alors

a) Deux solutions qui ne font pas partie du même e.r.m.m.i. sont indifférentes ou incomparables :

$$\forall\, (x, y) \in (MRM_1 \setminus MRM_2) \times (MRM_2 \setminus MRM_1), \quad x \simeq y \text{ ou } x \parallel y$$

b) Leur intersection est vide : $MRM_1 \cap MRM_2 = \varnothing$

Notons MRM, l'un des deux e.r.m.m.i..

c) Si card(MRM) ≥ 2 alors $\forall$ solution $x \in MRM$, $\exists\, y, z \in MRM \setminus \{x\}$ telles que $y \neq z$, $x \succ y$ et $x \prec z$
d) MRM a un cardinal toujours différent de 2.
e) Si S est fini et si card(MRM) > 2, alors il existe un circuit dans le sous-graphe partiel de $(S, \succcurlyeq)$ induit par MRM et $\succ$, contenant tous les sommets de MRM

Preuve : Une démonstration des assertions (a) et (b) est analogue à celles de la Proposition 2. Par conséquent nous n'en envisagerons pas ici.

(c) Raisonnons par l'absurde. Supposons qu'un tel couple (y, z) n'existe pas. Trois cas sont alors possible :
- toute solution t de MRM est incomparable ou indifférente à x. Par conséquent, $\{x\}$ est un e.r.m.m.i. strictement inclus dans MRM qui est aussi un e.r.m.m.i.. Ce qui contredit la définition d'un e.r.m.m.i.. Par conséquent, il existe un tel y ou un tel z.
- Supposons que $y \prec x$ existe, mais qu'il n'existe pas de $z \in MRM$ tel que $x \prec z$. Par conséquent, toute solution de t de MRM vérifie non$[t \succ x]$. Ce qui fait de $\{x\}$ un e.r.m.m.i. et contredit le fait que MRM soit aussi un e.r.m.m.i., car $x \in MRM$. Par conséquent un tel z existe.
- Supposons que $z \succ x$ existe, mais que y n'existe pas. Par conséquent, $\forall\, t \in MRM$, non$[x \succ t]$. Ce qui contredit le fait que MRM soit minimal par inclusion : $MRM \setminus \{x\}$ est un e.r.m.. Par conséquent un tel y existe.

En définitive, si card(MRM) ≥ 2 alors $\forall\, x \in MRM$, $\exists\, y, z \in MRM \setminus \{x\}$ telles que $y \neq z$, $x \succ y$ et $x \prec z$.

(d) Cette assertion est une conséquence de la proposition (c) : si MRM n'est pas réduit à un singleton $\{x\}$, alors tout x de MRM a un prédécesseur et un successeur tous deux disjoints.

(e) Une démonstration de cette assertion est analogue à celle de l'assertion (e) de la Proposition 2. Considérons le sous-graphe partiel de $(S, \succcurlyeq)$ induit par MRM et $\succ$, et noté $(MRM, \succ_{MRM})$, avec $\succ_{MRM} = \{(x, y) \in S$, tel que $x \in MRM$, $y \in MRM$ et $x \succ y\}$ et supposons que $2 <$ card(MRM) $\leq$ card(S) $< +\infty$.

Raisonnons par l'absurde : Supposons qu'il n'existe pas de circuit de longueur card(MRM) dans $(MRM, \succ_{MRM})$. Alors, d'après le Théorème 7 qu'il existe un cocircuit $\mathcal{C}$ dans $(MRM, \succ_{MRM})$. Notons $M_1(\mathcal{C})$ et $M_2(\mathcal{C})$ la bipartition de MRM effectuée par le cocircuit $\mathcal{C}$:

$M_1(\mathcal{C}) = \{z \in MRM$ tel que : $\exists\, t \in MRM$ et $(z, t) \in \mathcal{C}\}$ et $M_2(\mathcal{C}) = MRM \setminus M_1(\mathcal{C})$.

L'existence du cocircuit dans le graphe $(MRM, \succ_{MRM})$ est équivalent à dire que pour tout couple $(z, t) \in M_1(\mathcal{C}) \times M_2(\mathcal{C})$, non$[z \succ t]$. Or, MRM étant un e.r.m. de $(S, \succcurlyeq)$, non$[z \succ t]$ pour tout couple $(z, t) \in MRM \times (S \setminus MRM)$; et donc a fortiori pour tout couple $(z, t) \in M_1(\mathcal{C}) \times (S \setminus MRM)$, car $M_1(\mathcal{C}) \subset MRM$. En conséquence, pour tout couple $(z, t) \in M_1(\mathcal{C}) \times (S \setminus M_1(\mathcal{C}))$, non$[z \succ t]$. Ce qui signifie que $M_1(\mathcal{C})$ est un e.r.m. de $(S, \succcurlyeq)$, ce qui contredit le fait que MRM soit un e.r.m.m.i. ❑

Proposition 9 : Étant donné un ensemble non vide structuré par une relation binaire de préférence $(S, \succcurlyeq)$, alors

(a) PI($\succcurlyeq$) est une relation d'équivalence sur S
(b) x PI($\succcurlyeq$) $y \Rightarrow x$ et y sont indifférents
(c) Si $\succcurlyeq$ est transitive, alors (indifférence $\Leftrightarrow$ indiscernabilité parfaite).
(d) La transitivité de l'indifférence n'implique pas l'équivalence entre l'indiscernabilité parfaite et l'indifférence. En définitive, Si (indifférence $\Leftrightarrow$ indiscernabilité parfaite), alors $\succcurlyeq$ est transitive

Preuve : La preuve des trois premières assertions n'est pas présentée car elle est assez directe. Démontrons la validité de (d). Montrons d'abord que la transitivité de l'indifférence n'implique pas l'équivalence entre PI($\succcurlyeq$) et $\simeq$. Rappelons les tables de transitivité d'une relation binaire de préférence dont l'indifférence est transitive (Tableau 15).

T_{RBPIT}	$\simeq$	$\succ$	$\prec$	$\parallel$
$\simeq$	$\{\simeq\}$	$\{\succ, \prec, \parallel\}$	$\{\succ, \prec, \parallel\}$	$\{\succ, \prec, \parallel\}$
$\succ$	$\{\succ, \prec, \parallel\}$	$AF(RBP)$	$AF(RBP)$	$AF(RBP)$
$\prec$	$\{\succ, \prec, \parallel\}$	$AF(RBP)$	$AF(RBP)$	$AF(RBP)$
$\parallel$	$\{\succ, \prec, \parallel\}$	$AF(RBP)$	$AF(RBP)$	$AF(RBP)$

Tableau 15 : Table de transitivité des relations binaires dont l'indifférence est transitive.

La comparaison de cette table avec celle d'un préordre partiel (Tableau 14) montre qu'une relation binaire dont l'indifférence est transitive, n'est pas en général un préordre partiel. Si la relation d'indiscernabilité est équivalente à la relation d'indifférence, alors la table de transitivité $T_{\succcurlyeq}$ de $\succcurlyeq$ doit satisfaire la condition suivante : $T_{\succcurlyeq}(\alpha, \simeq) = \alpha$ pour tout $\alpha \in AF(RBP)$. Cette condition correspond à la première colonne de la table de transitivité d'un préordre partiel (Tableau 14). Elle démontre ainsi la première partie de l'assertion (d).
Montrons maintenant que l'équivalence entre PI($\succcurlyeq$) et $\simeq$ implique que $\succcurlyeq$ est transitive. La première colonne du Tableau 14 montre que l'indifférence est transitive ($\Leftrightarrow T_{\succcurlyeq}(\simeq, \simeq) = \simeq$). Il suffit alors de montrer que la préférence stricte est aussi transitive ($\Leftrightarrow T_{\succcurlyeq}(\succ, \succ) = \succ$). Remarquons que la première colonne du Tableau 14 montre aussi que $T_{\succcurlyeq}(\succ, \simeq) = \succ$. SEN [1970] montra l'équivalence[194] suivante :

Si ($T_{\succcurlyeq}(\simeq, \simeq) = \simeq$) et ($T_{\succcurlyeq}(\succ, \simeq) = \succ$) alors ($T_{\succcurlyeq}(\succ, \succ) = \succ$)

Ce qui démontre la transitivité de $\succcurlyeq$. ❑

[194] Amartya SEN appelle la transitivité de l'indifférence, la propriété II, la seconde condition, la propriété PI et la transitivité de la préférence stricte, la propriété PP.

4. Annexe B2 : Preuves de Propositions du Chapitre 6

Proposition 10 : Considérons une instance de problème de pure répartition $I_{PR} = (E, Q)$, avec $\mathrm{card}(E) = n > 2$. Alors, pour tout $x \in S(I_{PR})$, il existe une solution $y \in S(I_{PR})$ vérifiant $y \succ_{\mathrm{M}} x$.

Preuve : Construisons cette distribution y (Montrons que pour toute solution réalisable x ($\Leftrightarrow$ $\sum_{i=1}^{n} x(i) = Q$ et $x(i) \geq 0$ pour tout i), il est possible de construire une autre solution réalisable y ($\Leftrightarrow$ $\sum_{i=1}^{n} y(i) = Q$ et $y(i) \geq 0$ pour tout i), tel que $y \succ_{\mathrm{M}} x$).

Comme x est réalisable, il existe un indice k tel que $x(k) > 0$. Sélectionnons un sous-ensemble A de $E \setminus \{k\}$ tel que $\mathrm{card}(A) > n / 2$; et construisons y à partir de x, en distribuant la quantité $x(k)$ entre les individus de A. Récapitulons donc :

- $y(k) = 0$
- $y(i) = x(i) \qquad \forall\ i \in E \setminus (A \cup \{k\})$
- $y(i) = x(i) + \varepsilon_{x,i} > x(i) \ \ \forall\ i \in A$; avec $\sum_{i \in A} \varepsilon_{x,i} = x(i)$.

Ainsi, y est une solution réalisable, et le couple x, y vérifie par construction $CO(y \succ x) = A > n / 2$, $CO(x \succ y) = \{k\}$, et $CO(y \simeq x) = E \setminus (A \cup \{k\})$; avec $CO(y\ \alpha\ x)$ la coalition concordante avec la proposition "$y\ \alpha\ x$". Ainsi $y \succ_M x$. ❑

5. Annexe B3 : Preuves de Propositions du Chapitre 7

Proposition 12 : Le critère *VA* satisfait l'axiome d'égalité des proportions (EP) et l'invariance sous les permutations (IP). Par contre, *VA* n'est pas une extension du principe de transfert de Pigou-Dalton (PDT) pour $\mathrm{card}(E) > 2$, ni de la règle de Lorenz pour $\mathrm{card}(E) > 2$. En d'autres termes, pour x et y deux solutions de $S(I_{PR})$, si $y \succ_L x$ alors l'inégalité stricte $VA(x) > VA(y)$ n'est pas vérifiée.

Preuve :

(a) L'axiome EP est vérifié car la valeur minimale de la somme des déviations est zéro pour le vecteur de valeurs cibles x^*, tandis qu'elle est strictement positive dans les autres cas.

(b) Il vérifie l'axiome d'anonymat IP, comme conséquence du choix de poids tous égaux.

(c) Par contre, le principe de transfert de Pigou-Dalton n'est pas toujours respecté. Donnons un exemple : Considérons les distributions $x = (3, 6, 12)$ et $y = (4, 5, 12)$. La solution y est obtenue à partir de x en effectuant un transfert de Pigou-Dalton entre le premier et le second individu. Par conséquent, y est préféré à x selon ce principe. Or, le critère de (7–35) donne le même score 10 aux deux solutions. Par conséquent, ce critère ne satisfait pas toujours l'axiome PDT.

Effectuons maintenant la démonstration : Nous devons montrer que nous n'avons pas toujours $VA(x) > VA(y)$ lorsque y est obtenu à partir de x par PDT. Le principe de transfert de Pigou-Dalton nous indique qu'il existe un $\varepsilon > 0$ et un couple d'individus $i, j \in E$ tels que :

$$x(j) > y(i) = x(i) + \varepsilon \; ; \quad y(j) = x(j) - \varepsilon > x(i) \quad \text{et} \quad y(k) = x(k) \quad \text{pour tout } k \neq \{i, j\}$$

La valeur de *VA* aux points x et y dépend de la position de la moyenne $x^*(l) = Q / \mathrm{card}(E)$ par rapport à $x(l)$ et $y(l)$ pour $l \in \{i, j\}$ étant donné que le transfert se fait à moyenne égale. Le tableau ci-dessous récapitule cette relation.

$\boldsymbol{x^*(l)} \geq x(i) > y(l) > x(j)$ pour tout $l \in \{i, j\}$	$VA(x) = VA(y)$
$x(i) > y(l) > x(j) \geq \boldsymbol{x^*(l)}$ pour tout $l \in \{i, j\}$	
$x(i) > \boldsymbol{x^*(l)} > y(l) > x(j)$ pour tout $l \in \{i, j\}$	$VA(x) > VA(y)$
$x(i) > y(l) > \boldsymbol{x^*(l)} > x(j)$ pour tout $l \in \{i, j\}$	
$x(i) > y(i) > \boldsymbol{x^*(l)} > y(j) > x(j)$ pour tout $l \in \{i, j\}$	
$x(i) > y(j) > \boldsymbol{x^*(l)} > y(i) > x(j)$ pour tout $l \in \{i, j\}$	

Lorsque le transfert s'effectue entre deux individus moins bien lotis que la moyenne, ou au contraire mieux lotis, *VA* reste insensible. En conclusion :

$$y \text{ est obtenu à partir de } x \text{ par transfert de PDT} \Rightarrow VA(x) \geq VA(y) \qquad \text{(B3–54)}$$

Mais l'inégalité stricte n'est pas satisfaite.

Quand $\mathrm{card}(E) = 2$ avec $E = \{i, j\}$, seules les quatre dernières positions du tableau sont possibles, étant donné que par définition $x^*(l)$ est le barycentre de x et y :

$$x(i) + x(j) = y(i) + y(j) = x^*(i) + x^*(j) = Q.$$

D'où l'inégalité stricte.

(d) Ce critère n'est pas non plus une extension de la règle de Lorenz du fait qu'elle ne respecte pas le principe de transfert de Pigou-Dalton qui est sous-jacent à cette règle. De

plus, l'insensibilité de VA à certains transferts peut se généraliser : VA reste insensible aux transferts à moyenne égale entre plus de deux individus situés en amont ou en aval de la moyenne. ❑

Proposition 13 : Pour tout x, y de $S(I_{PR})$, si $x \succcurlyeq_L y$ alors $VA(x) \leq VA(y)$. La réciproque est fausse.

Preuve : Cette proposition est un corollaire de la proposition précédente et de l'implication (B3–54), étant donné que si x Lorenz-domine y alors il se décompose en une séquence finie non vide de transpositions et de transferts de Pigou-Dalton. ❑

Proposition 14 : Étant donnée une instance de problème de pure répartition I_{PR}, les assertions suivantes sont vraies :

4) Il existe des ensembles non vides $S \subseteq S(I_{PR})$ tels que : $B(S, (VA, \leq)) \setminus M(S, \succcurlyeq_L) \neq \varnothing$;
5) Il existe des ensembles non vides $S \subseteq S(I_{PR})$ tels que : $M(S, \succcurlyeq_L) \setminus B(S, (VA, \leq)) \neq \varnothing$;
6) Pour tout sous-ensemble fini non vide S de $S(I_{PR})$, nous avons :

$$M(S, \succcurlyeq_L) \cap B(S, (VA, \leq)) \neq \varnothing.$$

Preuve : Montrons les 2 premières assertions en exhibant des exemples :

1) Par exemple, pour $S = \{(3, 6, 12), (4, 5, 12)\}$ nous avons $B(S, (VA, \leq)) = S$ alors que $M(S, \succcurlyeq_L) = \{(4, 5, 12)\}$. Cet exemple montre même que l'on peut avoir $M(S, \succcurlyeq_L) \subset B(S, (VA, \leq))$.
2) Par exemple, pour $S = \{(3, 6, 12), (0, 10, 11)\}$ nous avons $B(S, (VA, \leq)) = \{(0, 10, 11)\}$ alors que $M(S, \succcurlyeq_L) = S$. Cet exemple montre aussi que l'on peut avoir $B(S, (VA, \leq)) \subset M(S, \succcurlyeq_L)$.
3) Raisonnons par l'absurde :

   Supposons qu'il existe un S non vide tel que l'intersection entre les deux ensembles soit vide. La définition de $M(S, \succcurlyeq_L)$ comme ensemble maximal nous dit que : pour toute solution $x \in M(S, \succcurlyeq_L)$, non $[y \succ_L x]$ pour tout $y \in S$.

   De plus, pour toute solution $y \in B(S, (VA, \leq)) \subseteq S \setminus M(S, \succcurlyeq_L)$, il existe un $x \in M(S, \succcurlyeq_L)$ tel que $x \succ_L y$. En effet, comme y n'est pas maximal, il existe un $z \in S \setminus \{y\}$ tel que $z \succ_L y$. Si $z \in M(S, \succcurlyeq_L)$, c'est gagné. Sinon, s'il existe une solution maximale x_1 telle que $x_1 \succ_L z$, alors par transitivité c'est gagné. Sinon, y est aussi incomparable à toute solution maximale x (selon la table de transitivité). De proche en proche, soit nous parvenons à une issue favorable, soit nous aboutissons à une impasse, lorsque toutes les solutions de $S \setminus M(S, \succcurlyeq_L)$ sont incomparables à toute solution maximale, et qu'il existe une chaîne de préférence stricte reliant les solutions non maximales pour Lorenz. La dernière solution non maximale n'étant pas Lorenz-dominé, sinon on aboutirait à une boucle, il est maximal. Ce qui contredit l'hypothèse de départ.

   Donc, il existe bien un $x \in M(S, \succcurlyeq_L)$ tel que $x \succ_L y$. Par conséquent, selon la Proposition 13, cela implique que $VA(y) \geq VA(x)$. Or y étant un élément optimal pour VA, pour tout $z \in S$, $VA(z) \geq VA(y)$. et donc à fortiori $VA(x) \geq VA(y)$.

   Par conséquent $VA(x) = VA(y)$, et $x \in B(S, (VA, \leq))$. Ce qui contredit l'hypothèse que l'intersection est vide. ❑

Proposition 26 : Considérons un septuplet $(V, D, I_e, \mathcal{A}, \succcurlyeq, \alpha, x)$ formé d'un ensemble de variables V, un ensemble de domaines D associé à V, un ensemble d'individus observés I_e, un sous-ensemble $\mathcal{A}$ du produit cartésien des domaines $\mathcal{D}_V$, une relation binaire de préférence étendue $\succcurlyeq$ sur $\mathcal{D}_V \times I_e$, une attitude α de $\mathcal{PR}(AF(RBP))$ et un élément x de $\mathcal{D}_V$. Notons $n = \text{card}(I_e)$.

Soit $A^{\mathcal{A}}y \geq b^{\mathcal{A}}$ un système d'inégalités linéaires telle que $A^{\mathcal{A}}$ est une matrice $m \times n^2$ à coefficients 0 ou 1, y un vecteur de dimension n^2 et b un vecteur entier de dimension m. En notant a_{kl} le coefficient de la k^e ligne et de la l^e colonne de la matrice $A^{\mathcal{A}}$, et $b^{\mathcal{A}}(k)$ le k^e coefficient du vecteur $b^{\mathcal{A}}$, Alors les deux assertions suivantes sont équivalentes :

(a) $A^{\mathcal{A}}y \geq b^{\mathcal{A}}$ est une description par des inégalités des facettes du polytope $P_{\mathcal{A}}(n)$

(b) $\mathcal{A} = \bigcap_{k=1}^{m} \bigcup_{\substack{B \subseteq \{l \mid a_{kl}=1\} \cap (U_1 \times U_2) \\ \text{et } \mathrm{card}(B)=b^{\mathcal{A}}(k)}} \bigcap_{e \in B} S(c_{\geqslant}[\alpha, x, i, j], V, D)$.

avec (i, j) le couple d'individus (c'est-à-dire de sommets) associé à l'arête e :

$$e = (u_1(i), u_2(j)) \in U_1 \times U_2 \text{ et } (i, j) \in I_e^2.$$

Preuve : Partons de l'assertion (a) et montrons l'équivalence (a) $\Leftrightarrow$ (b) : Considérons un vecteur entier de y_1 de $P_{\mathcal{A}}(n)$. Alors par définition $y_1 \in \{ z \in \quad^{n^2}$ tel que : $A^{\mathcal{A}}z \geq b^{\mathcal{A}} \}$ et y_1 est un vecteur de $\{0, 1\}^{n^2}$. Notons $a_{k\cdot}$ la k^e ligne de la matrice $\mathcal{A}$.

Traduisons une inégalité $a_{k\cdot} \cdot y_1 \geq b^{\mathcal{A}}(k)$:

Tout d'abord, les coefficients a_{kl} de la matrice $\mathcal{A}$ sont en 0 et 1, ainsi que les coefficients du vecteur y_1. Par conséquent,

$a_{k\cdot} \cdot y_1 \geq b^{\mathcal{A}}(k) \Leftrightarrow \exists$ au moins $b^{\mathcal{A}}(k)$ coefficients de y_1 égaux à 1 de manière à ce que l'inégalité soit respectée.

Ensuite, par définition y_1 est le vecteur d'incidence d'un graphe biparti $G = (U_1 \cup U_2, A)$ sur deux groupes de n sommets. Ce qui se traduit par :

$a_{k\cdot} \cdot y_1 \geq b^{\mathcal{A}}(k) \Leftrightarrow \exists$ au moins $b^{\mathcal{A}}(k)$ couples de sommets $(u_1, u_2) \in U_1 \times U_2$ tels que $y_1(e) = 1$, avec $e = (u_1, u_2)$, de manière à ce que l'inégalité soit respectée.

Comme G est par définition le graphe d'une solution $y_2 \in \mathcal{D}_V$ en (α, x) donc :

$a_{k\cdot} \cdot y_1 \geq b^{\mathcal{A}}(k) \Leftrightarrow \exists$ au moins $b^{\mathcal{A}}(k)$ couples d'individus $(i, j) \in I_e \times I_e$ tels que : $u_1 = u_1(i)$, $u_2 = u_2(j)$ et $y_2 \in S(c_{\geqslant}[\alpha, x, i, j], V, D)$, de manière à ce que l'inégalité soit respectée.

Maintenant en faisant le schéma inverse, et en écrivant plus formellement ces équivalences, nous obtenons :

$a_{k\cdot} \cdot y_1 \geq b^{\mathcal{A}}(k) \Leftrightarrow \exists$ au moins un sous-ensemble $B(I_e)$ de $I_e \times I_e$, de cardinal supérieur ou égal à $b^{\mathcal{A}}(k)$ tel que : $y_2 \in \bigcap_{(i,j) \in B(I_e)} S(c_{\geqslant}[\alpha, x, i, j], V, D)$, de manière à ce que l'inégalité soit respectée.

Utilisons à présent $U_1 \times U_2$ à la place de I_e^2, et remplaçons l'opérateur existentiel par son équivalent en termes d'opérateur ensembliste (i.e. l'union d'ensembles) :

$a_{k\cdot} \cdot y_1 \geq b^{\mathcal{A}}(k) \Leftrightarrow y_2 \in \bigcup_{\substack{B(G) \subseteq U_1 \times U_2 \\ \text{et } \mathrm{card}(B(G)) \geq b^{\mathcal{A}}(k)}} \bigcap_{(u_1(i), u_2(j)) \in B(G)} S(c_{\geqslant}[\alpha, x, i, j], V, D)$, de manière à ce que l'inégalité soit respectée.

Remarquons que s'il existe un sous-ensemble $B_1(G)$ de $U_1 \times U_2$, tel que $\mathrm{card}(B(G)) \geq b^{\mathcal{A}}(k)$, alors il est inclus dans le sous-ensemble $B_2(G)$ de $U_1 \times U_2$, tel que $\mathrm{card}(B(G)) = b^{\mathcal{A}}(k)$. Par conséquent, on peut se restreindre à l'égalité :

$a_{k\cdot} \cdot y_1 \geq b^{\mathcal{A}}(k) \Leftrightarrow y_2 \in \bigcup_{\substack{B(G) \subseteq U_1 \times U_2 \\ \text{et } \mathrm{card}(B(G)) = b^{\mathcal{A}}(k)}} \bigcap_{(u_1(i), u_2(j)) \in B(G)} S(c_{\geqslant}[\alpha, x, i, j], V, D)$, de manière à ce que l'inégalité soit respectée.

Maintenant, l'inégalité $a_{k\cdot} \cdot y_1 \geq b^{\mathcal{A}}(k)$ est respectée lorsque $B(G) \subseteq U_1 \times U_2$ et $\mathrm{card}(B(G)) = b^{\mathcal{A}}(k)$, mais aussi lorsque les arêtes choisies dans $U_1 \times U_2$ correspondent à des coefficients de la matrice $\mathcal{A}$ différents de 0. Nous obtenons par conséquent :

$$a_k \cdot y_1 \geq b^{\mathcal{A}}(k) \quad \Leftrightarrow \quad y_2 \in \bigcup_{\substack{B(G)\subseteq U_1\times U_2,\\ B(G)\subseteq\{l \mid a_{kl}=1\}\\ \text{et card}(B(G))=b^{\mathcal{A}}(k)}} \bigcap_{(u_1(i),u_2(j))\in B(G)} S(c_{\geqslant}[\alpha, x, i, j], V, D).$$

Après avoir traduit l'inégalité en union et intersection, sachant que y_1 vérifie simultanément toutes les inégalités décrivant le polytope $P_{\mathcal{A}}(n)$, nous obtenons le résultat :

$$\begin{cases} y_1 \in \{0,1\}^{4n^2} \text{ et} \\ A^{\mathcal{A}} y_1 \geq b^{\mathcal{A}} \end{cases} \Leftrightarrow \begin{cases} y_1 \in \{0,1\}^{4n^2} \text{ et} \\ y_1 \in P_{\mathcal{A}}(n) \end{cases} \Leftrightarrow y_2 \in \bigcap_{k=1}^{m} \bigcup_{\substack{B(G)\subseteq U_1\times U_2,\\ B(G)\subseteq\{l \mid a_{kl}=1\}\\ \text{et card}(B(G))=b^{\mathcal{A}}(k)}} \bigcap_{(u_1(i),u_2(j))\in B(G)} S(c_{\geqslant}[\alpha, x, i, j], V, D).$$

Ce qui prouve l'équivalence entre les deux assertions. ❑

Corollaire (du Théorème de Koënig-Hall) :

$$S(c_{\text{SSE}}[\alpha, x], V, D) = \bigcap_{\varnothing\neq J_1\subseteq I_e} \bigcup_{\substack{J_2\subseteq I_e \text{ tel que}\\ \text{card}(J_2)=\text{card}(J_1)}} \bigcap_{j\in J_2} \bigcup_{i\in J_1} S(c_{\geqslant}[\alpha, x, i, j], V, D).$$

Preuve : Partons de la description de $S(c_{\text{SSE}}[\alpha, x], V, D)$ par ses graphes valides (formule (7–49) page 249), et appliquons le théorème de Koënig-Hall :

$S(c_{\text{SSE}}[\alpha, x], V, D) = \{y \in \mathcal{D}_V$ tel que : $\text{card}(\{u_2 \in U_2(G_{\alpha,x}(y))$ vérifiant : $(u_1, u_2) \in A(G_{\alpha,x}(y))$ et $u_1 \in B\}) \geq \text{card}(B)$ pour tout $B \subseteq U_1(G_{\alpha,x}(y))\}$.

Rappelons que chaque sommet u d'un graphe de solution est associé à un unique individu k de I_e noté alors $u(k)$. Par conséquent :

$S(c_{\text{SSE}}[\alpha, x], V, D) = \{y \in \mathcal{D}_V$ tel que : $\text{card}(\{j \in I_e$ vérifiant : $(u_1(i), u_2(j)) \in A(G_{\alpha,x}(y))$ et $i \in J_1\}) \geq \text{card}(J_1)$ pour tout $J_1 \subseteq I_e\}$.

Ramenons-nous aux opérations ensemblistes. Par définition du graphe d'une solution, pour $(i, j) \in I_e^2$ donné, $(u_1(i), u_2(j)) \in A(G_{\alpha,x}(y)) \quad \Leftrightarrow \quad y \in S(c_{\geqslant}[\alpha, x, i, j], V, D)$.

Ensuite, pour un sous-ensemble $J_1 \subseteq I_e$ donné et un individu j donné, il existe un $i \in J_1$ tel que $(u_1(i), u_2(j)) \in A(G_{\alpha,x}(y))$ si et seulement si $y \in \bigcup_{i\in J_1} S(c_{\geqslant}[\alpha, x, i, j], V, D)$.

Deux sous-ensembles $J_1 \subseteq I_e$ et $J_2 \subseteq I_e$ donnés, tels que $\text{card}(J_2) \geq \text{card}(J_1)$, vérifient $y \in \bigcup_{i\in J_1} S(c_{\geqslant}[\alpha, x, i, j], V, D)$ pour tout $j \in J_2$, si et seulement si $y \in \bigcap_{j\in J_2} \bigcup_{i\in J_1} S(c_{\geqslant}[\alpha, x, i, j], V, D)$.

Par conséquent, pour un sous-ensemble $J_1 \subseteq I_e$ donné, il existe $J_2 \subseteq I_e$ tel que $\text{card}(J_2) \geq \text{card}(J_1)$, et $y \in \bigcap_{j\in J_2} \bigcup_{i\in J_1} S(c_{\geqslant}[\alpha, x, i, j], D)$, si et seulement si :

$$y \in \bigcup_{\substack{J_2\subseteq I_e \text{ tel que}\\ \text{card}(J_2)\geq\text{card}(J_1)}} \bigcap_{j\in J_2} \bigcup_{i\in J_1} S(c_{\geqslant}[\alpha, x, i, j], V, D).$$

Maintenant, il suffit de trouver un sous-ensemble J_2 tel que $\text{card}(J_2) = \text{card}(J_1)$, au lieu de chercher un J_2 vérifiant $\text{card}(J_2) \geq \text{card}(J_1)$, car s'il existe un ensemble vérifiant $\text{card}(J_2) \geq \text{card}(J_1)$, alors il en existe un vérifiant $\text{card}(J_2) = \text{card}(J_1)$.

Pour tout sous-ensemble non vide $J_1 \subseteq I_e$, il existe $J_2 \subseteq I_e$ tel que $\text{card}(J_2) \geq \text{card}(J_1)$, et $y \in \bigcap_{j\in J_2} \bigcup_{i\in J_1} S(c_{\geqslant}[\alpha, x, i, j], D)$, si et seulement si : $y \in \bigcap_{\varnothing\neq J_1\subseteq I_e} \bigcup_{\substack{J_2\subseteq I_e \text{ tel que}\\ \text{card}(J_2)=\text{card}(J_1)}} \bigcap_{j\in J_2} \bigcup_{i\in J_1} S(c_{\geqslant}[\alpha, x, i, j], V, D)$. Ce qui en définitive, revient à l'égalité entre les deux ensembles. ❑

Bibliographie

ABDENNADHER Slim & SCHLENKER Hans [1999] – INTERDIP : An Interactive Constraint Based Nurse Scheduler – *Proceedings of the 1rst International Conference and Exhibition on the Practical Application of Constraint Technologies and Logic Programming* (*PCALP'99*), Londres, avril 1999.

AICHHOLZER Oswin [1997] – *Combinatorial and Computational Properties of the Hypercube : New Results on Covering, Slicing, Clustering, and Searching on the Hypercube* – Thèse de l'Université de Technologie de Graz, Graz, Autriche, avril 1997.
Disponible à : http://www.cis.tu-graz.ac.at/igi/oaich.

AÏT YOUNES Amine, AZIBI Riad & ROY Bernard [2000*a*] – *Electre IS Manuel d'Utilisation Tome 1* – Document du LAMSADE n°118, Université Paris-Dauphine, mars 2000.

AÏT YOUNES Amine, AZIBI Riad & ROY Bernard [2000*b*] – *Electre IS Manuel d'Utilisation Tome 2* – Document du LAMSADE n°118 bis, Université Paris-Dauphine, juin 2000.

ARROW Kenneth J. [1951] – *Social Choice and Individual Values* – Cowles Foundations and Wiley, New-York.

ARROW Kenneth J. & INTRILIGATOR M. D. (eds.) [1986] – *Handbook of Mathematical Economics* – vol. 3, Elsevier Science Publishers B. V., North-Holland, 1986.

ARROW Kenneth J. & RAYNAUD Hervé [1986] – *Social Choice and Multicriterion Decision-Making* – the MIT Press, Cambridge, 1986.

ARROW Kenneth J., SEN Amartya Kumar & SUZUMURA Kotaro (eds.) [1997] – *Social Choice Re-Examined* – New-York, St Martin's Press, Macmillan, 1997.

ARROW Kenneth J., SEN Amartya Kumar & SUZUMURA Kotaro (eds.) [2002] – *Handbook of Social Choice and Welfare* – vol. 1, North Holland, *à paraître*.

D'ASPREMONT Claude & GEVERS Louis [2002] – *Social Welfare Functionals and Interpersonal Comparability – in* ARROW *et al*. [2002] : Chap. 10.

AYKIN Turgut [1996] – Optimal Shift Scheduling with Multiple Break Windows – *Management Science* 42 (4): 591-602.

AYKIN Turgut [2000] – A comparative Evaluation of Modeling Approaches to the Labour Scheduling Problem – *European Journal of Operational Research* 125 (2): 381-397.

BAKER Kenneth R. [1976] – Workforce Allocation in Cyclical Scheduling Problems : A Survey – *Operational Research Quarterly* 27 (1, ii) : 155-167.

BALAS Egon [1979] – Disjunctive Programming – *Annals of Discrete Mathematics* 5 : 3-51.

BANA E COSTA Carlos A. [1993] – *Les Problématiques dans le Cadre de l'Activité d'Aide à la Décision* – Document du LAMSADE n° 80, Université Paris-Dauphine, septembre 1993.

BANG-JENSEN Jørgen & GUTIN Grégory [2001] – *Digraphs : Theory, Algorithms and Applications* – coll. Springer monographs in mathematics, Springer-Verlag, Londres.

BARKER Paul (ed.) [1996] –*Living as Equals* – Oxford, Angleterre, Oxford University Press.

BARNETT William A., MOULIN Hervé, SALLES Maurice & SCHOFIELD Norman J. [1995] – *Social Choice, Welfare and Ethics* – Proceedings of the 8th International Symposium in Economic Theory and Econometrics, Cambridge University Press, Cambridge, 1995.

BASU K. [1987] – Axioms for Fuzzy Measures of Inequality – *Mathematical Social Sciences* 14 (3): 275-288.

BECHTOLD Stephen E. & JACOBS Larry W. [1990] – Implicit Modeling of Flexible Break Assignments in Optimal Shift Scheduling – *Management Science* 36 (11): 1339-1351.

BEAUMONT Nicolas [1997] – Using Mixed Integer Programming to Design Employee Rosters – *Journal of the Operational Research Society* 48 : 585-590.

BELDICEANU Nicolas [2000*a*] – *Global Constraints as Graph Properties on Structured Network of Elementary Constaints of the Same Type* – Technical Report T2000-01, Swedish Institute of Computer Science, 2000. réimpr. *in Lecture Notes in Computer Science* 1894, septembre 2000 : 52-66.

BELDICEANU Nicolas [2000*b*] – *Pruning for the Cardinality-Path Constraint Family* – Technical Report T2000-11, Swedish Institute of Computer Science, 2000.

BELMANDT Z. (nom collectif) [1993] – *Manuel de Prétopologie et ses Applications* – Éditions Hermès, Paris, France, chap. 16.

BENHAMOU Frédéric & OLDER William J. [1997] – Applying Interval Arithmetic to Real, Integer and Boolean Constraints – *Journal of Logic Programming* 32(1), juillet 1997 : 1-24.

BERGE Claude [1966] – *Espaces Topologiques, Fonctions Multivoques* – coll. Universitaire de mathématiques, Dunod, Paris.

BERGE Claude [1983] – *Graphes* – 3e éd., Gauthiers-Villars, Paris.

BESSIÈRE Christian, FREUDER Eugène C. & RÉGIN Jean-Charles [1995] – Using Inference to Reduce Arc Consistency Computation – *International Joint Conference on Artificial Intelligence* (*IJCAI'95*), Monréal, Canada, 1995, pp. 592-598.

BESSIÈRE Chistian, MESSEGUER Pedro, FREUDER Eugène C. & LARROSA Javier [1999] – On Forward-Checking for Non-binary Constraint Satisfaction – *Proceedings of IJCAI'99 Workshop on Non-binary constraints*, Stockholm, Suède : 9-16.

BESSIÈRE Christian & RÉGIN Jean-Charles [1997] – Arc Consistency for General Constraints Networks : Préliminary Results – *Proceedings of the International Joint Conference on Artificial Intelligence* (*IJCAI'97*), Nagoya, Japan, 1997, pp. 398-404.

BESSIÈRE Christian & RÉGIN Jean-Charles [1999] – Enforcing Arc-Consistency on Global Constraints by Solving Subproblems on the Fly – *CP'99* : *Lecture Notes in Computer Science* 1713 : 103-117.

BILLERA Louis J. & SARANGARAJAN A. [1996] – The Combinatorics of Permutations Polytopes – *in* BILLERA *et al*. [1996] : 1-23.

BILLERA Louis J., GREENE Curtis, SIMION Rodica & STANLEY Richard P. (éds.) [1996] – *Formal Power Series and Algebraic Combinatorics* – DIMACS series in discrete mathematics and theoretical computer science, vol. 24, The American Mathematical Society.

BILLIONET Alain [1999] – Integer Programming to Schedule a Hierarchical Workforce with Variable Demands – *European Journal of Operational Research* 114 : 105-114.

BISTARELLI Stefano, MONTANARI Ugo & ROSSI Francesca [1997] – Constraint Solving over Semirings – *Journal of the ACM* 44 (2) : 201-236.

BISTARELLI Stefano, MONTANARI Ugo, ROSSI Francesca, SCHIEX Thomas, VERFAILLIE Gérard & FARGIER Hélène [1999] – Semiring-Based CSPs and Valued CSPs: Frameworks, Properties, and Comparison – *Constraints : An International Journal* 4 (3) : 199-240.

BLACKORBY Charles, BOSSERT Walter & DONALDSON David [2002] – Utilitarianism and the Theory of Justice – *in* ARROW *et al*. [2002] chap. 11.

BLEUZEN-GUERNALEC Noëlle & COLMERAUER Alain [1997] – Narrowing a 2*n* Block of Sorting in O(*n*.log *n*) – *Lecture Notes in Computer Science* 1330, 1997 : 2-16.

BLEUZEN-GUERNALEC Noëlle & COLMERAUER Alain [2000] – Optimal Narrowing of a Block of Sortings in Polynomial Time – *Constraints : An International Journal* 5 (1-2), 2000 : 85-118.

BLIN J. M. [1976] – A Linear Assignment Formulation of the Multiattribute Decision Problem – *Revue Française d'Automatique d'Informatique et de Recherche Opérationnelle* 6 : 21-32.

BOISSONNAT J. D. & YVINEC M. [1995] – *Géométrie Algorithmique* – collection Informatique, Ediscience international, Paris, 1995.

BONMATI J. M. [1998] – *Les Ressources Humaines à l'Hôpital : Un Patrimoine à Valoriser* – collection « H », Berger-Levrault éd., avril 98.

BOURGUIGNON François [1979] – Decomposable Income Inequality Measures – *Econometrica* 47 (4), juillet 1979 : 901-920.

BOUTILIER Craig, BAFMAN Ronen I., GEIB Chris H. & POOLE David [1997] – A Constraint-Based Approach to Preference Elicitation and Decision Making – *in AAAI Spring Symposium on Qualitative Decision Theory*, Stanford : 19-28.

BOUTILIER Craig, BAFMAN Ronen I., HOOS Holger H. & POOLE David [1999] – Reasoning with Conditional Ceteris Paribus Preference Statements – *in Proceedings of the 15th Annual Conference on Uncertainty in Artificial Intelligence* (*UAI-99*), Stockholm : 71-80.

BOUYSSOU Denis, MARCHANT Thierry, PERNY Patrice, PIRLOT Marc, TSOUKIÀS Alexis & VINCKE Philippe [2000] – *Evaluation and Decision Models : a Critical Perspective* – Kluwer Academic, Dordrecht, 288 pp.

BOUYSSOU Denis & PIRLOT Marc [1997] – Conjoint Measurement Without Additivity and Transitivity – *Proceedings of the International Conference on Methods and Applications of Multicriteria Decision Making*, Mons, 14-16 mai 1997, 165-170.

BRANDSTÄDT Andreas, LE Van Bang & SPINRAD Jeremy P. [1999] – *Graph Classes. A Survey* – SIAM Monographs on Discrete Mathematics and Applications, Society for Industrial and Applied Mathematics, chap. 6 Posets : 91-103.

BRUSCO Michael J. & JACOBS Larry W. [2000] – Optimal Models for Meal-Break and Start-Time Flexibility in Continuous Tour Scheduling – *Management Science* 46 (12): 1630-1641.

BRUSCO Michael J. & JOHNS Tony R. [1996] – A Sequential Integer Programming Method for Discontinuous Labor Tour Scheduling – *European Journal of Operational Research* 95: 537-548.

BULATOV Andrei A., KROKHIN Andrei A. & JEAVONS Peter [2000] – Constraint Satisfaction Problems and Finite Algebras – *Lecture Notes in Computer Science* 1853, Springer-Verlag : 272-282.

CAI X. & LI K. N. [2000] – A Genetic Algorithm for Scheduling Staff of Mixed Skills under Multi-Criteria – *European Journal of Operational Research* 125 (2), septembre 2000 : 359-369.

CARR Robert, FUJITO Toshihiro, KONJEVOD Goran & PAREKH Ojas [2000] – A $2\frac{1}{10}$-Approximation Algorithm for a Generalization of the Weighted Edge-Dominating Set Problem – *in Lecture Notes in Computer Science* 1879 (*ESA 2000*), Spinger : 132-142.

CASEAU Yves, GUILLO Pierre-Yves & LEVENEZ Eric [1995] – A Deductive and Object-Oriented Approach to a Complex Scheduling Problem – *Journal of Intelligent Information Systems* 4 : 149-166.

CHAN Peter [2002] – *La Planification du Personnel : Acteurs, Actions et Multiples pour une Planification Opérationnelle des Personnes* – Thèse de l'Université Joseph Fourier, Grenoble I, Laboratoire TIMC, octobre 2002.

CHAN Peter & WEIL Georges [2000] – Cyclical Staff Scheduling Using Constraint Logic Programming – *Lecture Notes in Computer Science* 2079, Springer-Verlag : 159-175.

CHARNES A. & COOPER W. W. [1977] – Goal Programming and Multiple Objective Optimizations – *European Journal of Operational Research* 1 : 39-54.

CHARTRAND Gary & LESNIAK Linda [1996] – *Graphs and Digraphs* – 3e édition, Chapman & Hall, Londres.

CHENG B. M. W., CHOI K. M. F., LEE J. H. M., WU J. C. K. [1999] – Increasing Constraint Propagation by Redundant Modeling : an Experience Report – *Constraints* 4 (2), mai 1999 : 167-192.

CHENG B. M. W., LEE J. H. M., WU J. C. K. [1997] – A Nurse Rostering System Using Constraint Programming and Redundant Modeling – *IEEE Transactions in Information Technology in Biomedecine* 1 (1) : 44-54.

COOPER Martin C. [1989] – An Optimal *k*-Consistency Algorithm – *Artificial Intelligence* 41 : 89-95.

CORNER James, BUCHANAN John & HENIG Mordecai [2001] – Dynamic Decision Problem Structuring – *Journal of Multi-Criteria Decision Analysis* 10, 2001 : 129-141.

COSYTEC (ed.) [1995] – *CHIP C Library* – COSITEC SA, Parc Club Orsay Université ; 4, rue Jean Rostand ; 91893 Orsay cedex, France.

COTÉ Murray J. & STEIN William E. [2000] – An Erlang-based Stochastic Model for Patient Flow – *Omega* 28 : 347-359.

COURBON Jean Claude, DUBOIS Didier, POMEROL Jean-Charles & ROY Bernard [1994] – *Autour de l'Aide à la Décision et de l'Intelligence Artificielle* – Rapport de recherche Laforia/IBP 1994/01, Institut Blaise Pascal, Université Paris 6, Paris, France, conférence du 10 juin 1993, 25 pages.
Disponible à : http://www.lip6.fr/reports/.

CRÈTE Jean & IMBEAU Louis M. [1996] – *Comprendre et Communiquer la Science* – coll. Méthodes en Sciences Humaines, 2e édition revue et corrigée, De Boeck Université, Bruxelles.

DAGO Pierre & VERFAILLIE Gérard [1996] – Nogood Recording for Valued Satisfaction Problems – *in Proceedings of the 8th IEEE International Conference on Tools with Artificial Intelligence* (*ICTAI'96*), Toulouse, France : 132-139.

DAHL Geir [1999*a*] – Matrix Majorization – *Linear Algebra and its Applications* 288 : 53-73 ; réimp. Rapport technique 263, Université d'Oslo, département d'informatique, Norvège, juin 1998, 20 pages.

DAHL Geir [1999*b*] – Majorization Polytopes – *Linear Algebra and its Applications* 297 : 157-175.

DANTZIG Georges B. [1954] – A Comment on Edie's "Traffic Delays at Toll Booths" – *Operations Research* 2 (3): 339-341.

DASGUPTA Partha, SEN Amartya Kumar & STARRETT David [1973] – Notes on the Measurement of Inequality – *Journal of Economic Theory* 6 : 180-187.

DECHTER Rina & VAN BEEK Peter [1995] – Local and Global Relational Consistency – *CP'95* : *Lecture Notes in Computer Science* 976 : 240-257.

DEMONGEOT Jacques, LE BEUX Pierre & WEIL Georges (eds.) [1994] – *Informatisation de l'unité de soins du futur* – coll. Informatique et santé, vol. 7, Springer-Verlag, Paris, France.
Disponible à : http://www.hbroussais.fr/Broussais/InforMed/InforSante/Volume7/TexteVol7.html.

DENICOLÓ Vincenzo [1999] – No-envy and ARROW's conditions – *Social Choice and Welfare* 16 (4) : 585-597.

DUBOIS Didier & PRADE Henri [2002] – La Problématique Scientifique du Traitement de l'Information – *Revue Information-Interaction-Intelligence* 1(2).
Disponible à : http://www.revue-i3.org/.

DURAND Sylvain [2000] – *Sur Quelques Paradoxes en Théorie du Choix Social et en Décision Multicritère* – Thèse de l'Université Joseph Fourier, Grenoble, France.

EISENBRAND Friedrich, RINALDI Giovanni & VENTURA Paolo [2002] – 0/1 Optimization and 0/1 Primal Separation are Equivalent – *in Proceedings of the 13th Annual ACM-SIAM Symposium on Discrete Algorithms* (*SODA 2002*), January 6-8, 2002, San Francisco, CA, USA. ACM/SIAM : 920-926.

EITER Thomas & GOTTLOB Georg [1995] – Identifying the Minimal Transversals of a Hypergraph and Related Problems – *SIAM Journal on Computing* 24 (6), décembre 1995 : 1278-1304.

Équipe MUSTIL [1996]– *Sur la Notion de Modèle en Informatique* – rapport de recherche Laforia 1996/12, Institut Blaise Pascal, Université Paris 6, Paris, France, octobre 1996, 17 pages.
Disponible à : http://www.lip6.fr/reports/.

EHRGOTT Matthias & GANDIBLEUX Xavier [2000] – *An Annotated Bibliography of Multiobjective Combinatorial Optimization* – rapport de recherche n°62/2000, Université Kaiserslautern, Allemagne ; réimpr. *OR Spektrum* 22, 2000 : 425-460.

EHRGOTT Matthias [2000] – Approximation Algorithms for Combinatorial Multicriteria Optimization Problems – *International Transactions in Operational Research* 7, pp. 5-31.

EMELICHEV V. A. & PEREPELITSA V. A. [1994] – Complexity of Discrete Multicriteria Problems – *Discrete Mathematics and Applications* 4 (2), pp. 89-117.

FAGES François [1996] – *Programmation Logique par Contraintes* – Ellipses-Marketing, Paris, 183 p.

FALTINGS Boi [1994] – Arc-Consistency for Continuous Variables – *Artificial Intelligence* 65(2) : 363-376.

FERBER Jacques [1995] – *Les Systèmes Multi-Agents. Vers une Intelligence Collective* – InterEditions, Paris.

FERLAND Jacques A. [1998] – Generalized Assignment-Type Problems as a Powerful Modelling Scheme – *Lecture Notes in Computer Science* 1408 : 53-77.

FERLAND Jacques A., HERTZ Alain & LAVOIE Alain [1996] – An Object-Oriented Methodology for Solving Assignment Type Problems with Neighborhood Search Techniques – *Operations Research* 44 (2), mars-avril 1996 : 347-359.

FERLAND Jacques A. & LAVOIE Alain [1992] – Exchange Procedures For Timetabling Problems – *Discrete Applied Mathematics* 35 : 237-243.

FERLAND Jacques A. & MARZILIANO Pina [2001] – Efficient and Local Efficient Solutions for Assignment Type Problems – *RAIRO Operations Research* 35 (3) : 301-313.

FISHBURN Peter C. [1987] – *Interprofil Conditions and Impossibility* – Harwood Academic Publishers, New York.

FLOUDAS Christodoulos A. [1995] – *Nonlinear and Mixed-Integer Optimization. Fundamentals and Applications* – coll. Topics in chemical engineering, Oxford University Press, Oxford.

FODOR J. & ROUBENS Marc [1994] – *Fuzzy Preference Modelling and Multicriteria Decision Support* –Kluwer Academic, Dordrecht.

FOSTER James E. & OK Efe A. [1999] – Lorenz Dominance and the Variance of Logarithms – *Econometrica* 67 (4), juillet 1999 : 901-907.

FOSTER James E. & SEN Amartya Kumar [1997] – On Economic Inequality After a Quarter Century – *in* SEN [1973] annexe substantielle de l'édition élargie de 1997.

FREUDER Eugène C. [1978] – Synthetizing Constraints Expressions – *Communication of the ACM* 21 : 958-966.

FREUDER Eugène C. [1982] – A Sufficient Condition for Backtrack-Free Search – *Journal of the ACM* 29(1), 1982 : 24-32.

FREUDER Eugène C. [1985] – A Sufficient Condition for Backtrack Bounded Search – *Journal of the ACM* 32(4), 1985, 755-761.

FREUDER Eugène C. & WALLACE Rick J. (eds.) [2001] – *Constraint Programming and Large Scale Optimisation Problems* – coll. DIMACS Series, vol. 57, American Mathematical Society.

FROST Daniel Hunter [1997] – *Algorithms and Heuristics for Constraint Satisfaction Problems* – Thèse de l'Université de Californie, Irvine.

GARBAY Catherine [2003] – Les Sciences du Traitement de l'Information comme Pivot de l'Interdisciplinarité : une Vision Systémique – *Revue Information-Interaction-Intelligence* 2(2).
Disponible à : http://www.revue-i3.org/.

GAREY Michael R. & JOHNSON David S. [1979] – *Computers and Intractability : A Guide to the Theory of NP-Completeness* – W. H. Freeman and Company, New-York, 1979.

GASCON Viviane, VILLENEUVE Sophie, MICHELON Philippe & FERLAND Jacques A. [2000] – Scheduling the Flying Squad Nurses of a Hospital Using a Multi-Objective Programming Model – *Annals of Operations Research* 96 : 149-166.

GAUVIN Annie & JACOT Henri [1999] – *Temps de Travail, Temps Sociaux, pour une Approche Globale* – coll. Liaisons Sociales, éditions Liaisons.

GAVANELLI Marco [2001] – Partially Ordered Constraint Optimization Problems – *LNCS 2239*, *CP'2001*, Springer-Verlag, Berlin, 2001, page 763.

GAVANELLI Marco [2002*a*] – *Interactive Constraint Satisfaction Problems for Artificial Vision* – Thèse de l'Université de Ferrara, Italie, janvier 2002.

GAVANELLI Marco [2002*b*] – An Implementation of Pareto Optimality in CLP(FD) – *Proceedings of CP-AI-OR'02*, : 49-64.

GHOSHAL J., LASKAR R. & PILLONE D. [1998] – Topics on Domination in Directed Graphs – *in* HAYNES *et al*. [1998*b*], chapitre 15 : 401-437.

GONDRAN Michel & MINOUX Michel [1995] – *Graphes et Algorithmes* – coll. de la Direction des Études et Recherches d'Électricité de France, éditions Eyrolles, 3e édition revue et augmentée.

GRECO Salvatore, MATARAZO Benedetto & SLOWINSKI Roman. [2001] – Rough Set Theory for Multicriteria Decision Analysis – *European Journal of Operational Research* 129 : 1-47.

GREENBERG Harvey J. [1998] – An Annotated Bibliography for Post-solution Analysis in Mixed Integer Programming and Combinatorial Optimization – *in* WOODRUFF [1998] : 97-148.
Disponible à : http://carbon.cudenver.edu/~hgreenbe/.

GUITOUNI Adel, MARTEL Jean-Marc & VINCKE Philippe [2000] – A Framework to Choose a Discrete Multicriterion Aggregation Procedure – Rapport de recherche n°2000/03, Université Libre de Bruxelles, Belgique.

HAMMOND Peter J. [1979] – Equity in Two Person Situations: Some Consequences – *Econometrica* 47 (5), septembre 1979 : 1127-1135.

HAMMOND Peter J. [1991] – Independence of Irrelevant Interpersonal Comparisons – *Social Choice and Welfare* 8 : 1-19.

HAMMOND Peter J. [1998] – Interpersonally Comparable Utility – *in* BARBERÀ Salvador and SEIDL Christian (eds.) [1998] – *Handbook of Utility Theory* – Vol. 2: Extensions, Kluwer Academic Publishers : chapitre 21.

HANSEN Pierre, JAUMARD Brigitte & VOVOR Tsevi [1998] – Solving the Bicriterion Shortest Path Problem from both Ends – *Les Cahiers du GERAD* G-98-17, mai 1998, 19 p.

HAYNES Teresa, HEDETNIEMI Stephen & SLATER Peter J. [1998*a*] – *Fundamentals of Domination in Graphs* – coll. Monographs and textbooks in pure and applied mathematics, vol. 208, Marcel Dekker, New-York.

HAYNES Teresa, HEDETNIEMI Stephen & SLATER Peter J. [1998*b*] – *Domination in Graphs. Advanced Topics* – coll. Monographs and textbooks in pure and applied mathematics, vol. 209, Marcel Dekker, New-York.

HENNING Michael A. [1996] – Domination in Graphs : A Survey – *Congressus Numerantium* 116 : 139-172.

HERTZ Alain, ROBERT Vincent & BERTHOD Vincent [2000] – Planification des Emplois du Temps et de la Formation au sein d'une Grande Entreprise – *RAIRO Operations Research* 34(1) : 61-83.

HEUS Kamel [1996] – *Gestion des Plannings Infirmiers, Application des Techniques de Programmation par Contraintes* – Thèse de l'Université Joseph Fourier, Grenoble I, Laboratoire TIMC, mai 1996.

HICKEY Timothy J. [2001] – Métalevel Interval Arithmetic and Verifiable Constraint Solving – *Journal of Functional and Logic Programming* 2001 (7).

HICKEY Timothy J., JU Qun & VAN EMDEN Maarten H. [2001] – Interval Arithmetic: from Principles to Implementation – *Journal of the ACM* 48 (5), septembre 2001, pp. 1038-1068.

ILOG (éd.) [1995] – *ILOG SOLVER : Object Oriented Constraint Programming* – ILOG SA ; 12, avenue Raspail ; BP 7 ; 94251 Gentilly cedex ; France.

JACQUET-LAGRÈZE Eric, MONTAUT Denis & PARTOUCHE Ariane [1998] – The Shift Scheduling Problem: Different Formulations and Solution Methods – *Foundations of Computing and Decision Sciences* 23 (4): 199-217.

JASZKIEWICZ Andrzej [1997] – A Metaheuristic Approach to Multiple Objective Nurse Scheduling – *Fundations of Computing and Decision Science* 22 (3) : 169-184.

JASZKIEWICZ Andrzej [2001] – *Multiple Objective Metaheuristic Algorithms for Combinatorial Optimization* – Thèse d'Habilitation n°330, Université de Technologie de Poznań, Poznań, Pologne.

JAULIN Luc, KIEFFER Michel, DIDRIT Olivier & WALTER Eric [2001] – *Applied Interval Analysis, with Examples in Parameter and State Estimation, Robust Control and Robotics* – Springer-Verlag, Berlin.

JAUMARD Brigitte, SEMET Fédéric & VOVOR Tsevi [1998] – A Generalized Linear Programming Model for Nurse Scheduling – *European Journal of Operational Research* 107 (1): 1-18.

JEAVONS Peter G. [1998] – On the Algebraic Structure of Combinatorial Problems – *Theoretical Computer Science* 200 :185-204.

JEAVONS Peter G., COHEN D. A. & GYSSENS M. [1995] – A Unifying Framework for Tractable Constraints – *Lecture Notes in Computer Science* 976, Springer-Verlag : 276-291.

JOHNSON David S., YANNAKAKIS Mihalis & PAPADIMITRIOU Christos H. [1988] – On Generating All Maximal Independent Sets – *Information Processing Letters* 27 : 119-123.

JOSEPH Rémy-Robert [1998] – *Conception d'un Logiciel d'Aide à l'Élaboration des Plannings de Personnel en utilisant la Programmation Par Contraintes Orientée Objet : Modélisation et Contraintes Génériques* – Rapport de DEA de recherche opérationnelle, Institut National Polytechnique de Grenoble, Rapport interne de la société Equitime, 1998.

JOSEPH Rémy-Robert, CHAN Peter & WEIL Georges [2002] – Choix Individuels, Équité et Systèmes Informatiques d'Aide à l'Élaboration d'Horaires de Travail – *Actes du 4e Congrès de la Société Française de Recherche Opérationnelle et d'Aide à la Décision* (*ROADEF 2002*) : p. 169.

JOSEPH Rémy-Robert, CHAN Peter, HIROUX Michael & WEIL Georges [2003*a*] – Preference Constraints: New Global Soft Constraints Dedicated to Preference Binary Relations (Summary) – *CP 2003 : Lecture Notes in Computer Science* 2833, Springer-Verlag, Berlin : p. 977.

JOSEPH Rémy-Robert, CHAN Peter, HIROUX Michael & WEIL Georges [2003] – Preference Constraints: New Global Soft Constraints Dedicated to Preference Binary Relations – *Proceedings of the 2nd International Workshop on Modelling and Reformulating Constraint*

Satisfaction Problems: Towards Systematisation and Automatisation, Kinsale, Irlande, 29 septembre 2003 : 104-109.

JUSSIEN Narendra [1997] – *Relaxation de Contraintes pour les Problèmes Dynamiques* – Thèse de l'Université de Rennes 1, Rennes, France.

JUSSIEN Narendra & BOIZUMAULT Patrice [1997] – Stratégies en Meilleur d'Abord pour la Relaxation de Contraintes – *Journées Francophones de Programmation Logique par Contraintes : JFPLC*'97, 149-165.

KEENEY Ralph L. & RAÏFFA Howard [1976] – *Decisions with Multiple Objectives: Preferences and Value Tradeoffs* – John Wiley & Sons, New York.

KEENEY Ralph L. [1988] – Structuring Objectives for Problems of Public Interest – *Operations Research* 36 (3) : 396-405.

KOBEISSI Mohamed [2001] – *Prolongement de Graphes dans l'Hypercube* – Thèse de l'Université Joseph Fourier, Grenoble, France.

KÖHLER G. [1978] – *Choix Multicritère et Analyse Algébrique des Données Ordinales* – Thèse de 3e cycle, Université Scientifique et Médicale de Grenoble, France.

KOSHEVOY Gleb [1997] – The Lorenz Zonotope and Multivariate Majorizations – *Social Choice and Welfare* 15 (1) : 1-14.

KOSHEVOY Gleb [1999] – *Price Majorization and the Inverse Lorenz Function* – Rapport de recherche n°3/99, Université de Cologne (Köln), Allemagne, septembre 1999 : 27 pages.

KOSTREVA Michael M. & OGRYCZAK Włodzimierz [1999] – Linear Optimization with Multiple Equitable Criteria – *RAIRO Operations Research* 33 : 275-297.

KRISHER Richard [2000] – Putting the Power of Time in Nurses' Hands – *Advance for Nurse*, 8 mai 2000, page 22.
Disponible à : http://www.advancefornurses.com/promo/resourceguide.asp?doc=may8_00p22.

KRIVINE Jean-Louis [1998] – *Théorie des Ensembles* – coll. Nouvelle bibliothèque mathématique 5, Paris, Cassini.

LAFAGE Céline & LANG Jérôme [2001] – Propositional Distances and Preference Representation – *Lecture Notes in Artificial Intelligence* 2143, Springer-Verlag : 48-59.

LAHIRI Somdeb [2002] – Basic Quasirationality – présenté à *IFORS 2002*, 7 p..

LANG Jérôme [2002] – De la Représentation des Préférences au Vote Combinatoire. Un État de l'Art et une Étude de Complexité – *Actes du 13ème Congrès Francophone AFRIF-AFIA de Reconnaissance de Formes et d'Intelligence Artificielle*, Centre de Congrès d'Angers, 8-10 janvier 2002 : 945-954.

LAPORTE Gilbert [1999] – The Art and Science of Designing Rotating Schedules – *Journal of the Operational Research Society* 50 (10) : 1011-1017.

LÉVINE Pierre & POMEROL Jean-Charles [1989] – *Systèmes Interactifs d'Aide à la Décision et Systèmes Experts* – coll. Traité des nouvelles technologies, série décision assistée par ordinateur, Hermès, Paris.

LI Ninghua & LI Ling X. [2000] – Modelling Staffing Flexibility : A Case of China – *European Journal of Operational Research* 124 : 255-266.

LIN C. K. Y., LAI K. F. & HUNG S. L. [2000] – Development of a Workforce Management System for a Customer Hotline Service – *Computers and Operations Research* 27 : 987-1004.

LIST Chistian [1999] – Multidimensional Inequality Measurement : A Proposal – *Nuffield College Working Paper in Economics*, novembre 1999.

MAASOUMI Esfandiar [1986] – The Measurement and Decomposition of Multi-Dimensional Inequality – *Econometrica* 54 (4), juillet 1986 : 991-997.

MACKWORTH Alan K. [1977] – Consistency in Networks of Relations – *Artificial Intelligence* 8 : 99-118.

MAC MURRIN S. M. (éd.) [1980] – *Tanner Lectures on Human Values I* – Salt Lake City, University of Utah Press, et Cambridge, Cambridge University Press.

MANDLER Michael [2001] – *Compromise between Cardinality and Ordinality in Preference Theory and Social Choice* – Rapport de recherche n°1322, Cowles Fundation, Université de Yale, New Haven, Connecticut, Etats-Unis, aout 2001, 41 pages.

MARCHANT Thierry, BOUYSSOU Denis, PERNY Patrice, PIRLOT Marc, TSOUKIÀS Alexis & VINCKE Philippe [2000] – Choosing on the Basis of Several Viewpoints : the Example of Voting – *in* BOUYSSOU *et al*. (eds) [2000].

MARCHANT Thierry & BOUYSSOU Denis [1999] – A Common Framework for Describing Most Aggregation Procedures in Decision Aiding – Rapport de recherche n°99/23, Université Libre de Bruxelles, Belgique.

MARCOTTE Odile & SOLAND Richard M. [1986] – An Interactive Branch-and-Bound Algorithm for Multiple Criteria Optimization – *Management Science* 32(1), janvier 1986 : 61-75.

MARSHALL Albert W. & OLKIN Ingram [1979] – *Inequalities : Theory of Majorization and Its Applications* – coll. Mathematics in Science and Engineering, vol. 143, Academic Press, New-York.

MARATHE Madhav V. [2002] – Towards a Predictive Computational Complexity Theory – *Lecture Notes in Computer Science* 2380, Springer-Verlag, Berlin : 22-31.

MARTINEZ David & VERFAILLIE Gérard [1998] – Une Calculette à Contraintes pour la Résolution Interactive de CSP – *Actes des 4es Journées Nationales de Programmation Par Contraintes JNPC'98* : 13-20.

MEHLHORN Kurt & THIEL Sven [2000] – Faster Algorithms for Bound-Consistency of the Sortedness and the Alldifferent Constraint – *Lecture Notes in Computer Science* 1894, Springer, 2000 : 306-319.

MEYER AUF'M HOFE Harald [2001] – Nurse Rostering as Constraint Satisfaction with Fuzzy Constraints and Inferred Control Strategies – *in* FREUDER & WALLACE [2001] : 67-99.

MEYER AUF'M HOFE Harald [2002] – Global Constraints with Fuzzy Extensions – rapport technique du SIEDA w-2002/1.

MIETUS Dirkje Magrieta [1994] – *Understanding Planning for Effective Decision Support ; A Cognitive Task Analysis of Nurse Scheduling* – Thèse de la Research Institute on Systems, Organisations and Management de l'Université de Groningen, labyprint publication.
Disponible à : http://docserver.ub.rug.nl/eldoc/dis/management/.

MILLER Holmes E. & PIERSKALLA William P. & RATH Gustave J. [1976] – Nurse Scheduling Using Mathematical Programming – *Operations Research* 24 (5), septembre-octobre 1976 : 857-870.

MINOUX Michel [1998] – *Résolution de Systèmes Linéaires dans les Semi-Anneaux et les Dioïdes* – Rapport de recherche LIP6 1998/051, Institut Blaise Pascal, Université Paris 6, Paris, France.

MINTZBERG Henry [1998] – *Structure et Dynamique des Organisations* – Les Editions d'Organisation, 12e tirage, Paris.

MOHR Roger & HENDERSON Thomas C. [1986] – Arc and Path Consistency Revisited – *Artificial Intelligence*, 28 (2), 1986, pp. 225-233.

MOHR Roger & MASINI Gérald [1988] – Good Old Discrete Relaxation – *ECAI'88*, Munchen, 1988, pp. 651-656.

MONTANARI Ugo [1974] – Networks of Contraints : Fundamental Properties and Applications to Picture Processing – *Information Sciences* 7 : 95-132.

MOONDRA S. L. [1976] – A L.P. Model for Workforce Scheduling for Banks – *Journal of Bank Research* 6 (4): 299-301.

MORIN Jean-Michel [1999] – *Sociologie de l'Entreprise* – Que sais-je ? vol. 3493, 1ère édition, Paris, nov. 1999.

MOULEN Joel, ANDJIGA Nicolas Gabriel & BRISSAUD Marcel [1999] – *Préférences Individuelles et Choix Collectifs* – coll. Interdisciplinarité et nouveaux outils, Hermès science publications.

MOULIN Hervé [1988] – *Axioms of Cooperative Decision Making* – coll. Economic society monographs n° 15, Cambridge university press, Cambridge.

MOULIN Hervé [1995] – *Cooperative Economics : A Game-Theoric Introduction* – Princeton university press, Princeton.

MOULIN Hervé & THOMSON William [1997] – *Axiomatic Analysis of Ressouce Allocation Problems – in* ARROW *et al*. [1997] : 101-120.

MOUSSEAU Vincent [1992] – Analyse et Classification de la Littérature Traitant de l'Importance Relative des Critères en Aide Multicritère à la Décision – Cahier du LAMSADE n°109, Université de Paris-Dauphine, février 1992.

MUSLIU Nysret [2001] – *Intelligent Search Methods for Workforce Scheduling : New Ideas and Practical Applications* – Thèse de l'Université Technique de Vienne (Wien), Autriche, septembre 2001.

NARASIMHAN Rangarajan [2000] – An Algorithm for Multiple Shift Scheduling of Hierarchical Workforce on Four-Day or Three-Day Workweeks – *INFOR* 38 (1) : 14-32.

NEMHAUSER George L. & WOLSEY Laurence A. [1988] – *Integer and Combinatorial Optimization* – Series in Discrete Mathematics and Optimization, John Wiley and sons, New-York.

NEWELL A. & SIMON Herbert A. [1972] – *Human Problem Solving* – Prentice Hall, Englewood Cliffs.

NIELSEN Søren R., PISINGER David & MARQUARDSEN Per [2000] – Automatic Transformation of Constraint Satisfaction Problems to Integer Linear Form – An Experimental Study – *Conférence CP2000, atelier TRICS (Techniques for implementing constraint programming systems)*, 22 septembre 2000, Hilton, Singapour.
Disponible à : http://www.comp.nus.edu.sg/~cp2000/trics/.

OGRYCZAK Włodzimierz [2000] – Inequality measures and Equitable Approaches to Location Problems – *European Journal of Operational Research* 122 : 374-391.

OK Effe A. [1995] – Fuzzy Measurement of Income Inequality: A Class of Fuzzy Inequality Measures – *Social Choice and Welfare* 12 (2) : 111-136.

OLDENKAMP Johan Hendrick [1996] – *Quality in Fives ; On the Analysis, Operationalization and Application of Nursing Schedule Quality* – Thèse de la Research Institute on Systems, Organisations and Management de l'Université de Groningen, labyprint publication.
Disponible à : http://docserver.ub.rug.nl/eldoc/dis/management/.

OTHMANI Imed [1998] – *Optimisation Multicritère : Fondements et Concepts* – Thèse de l'Université Joseph Fourier, Grenoble, France.

PADBERG Manfred [1995] – *Linear Optimization and Extensions* – série Algorithms and Combinatorics 12, Springer, Berlin.

PAPADIMITRIOU Christos H. [1994] – *Computational Complexity* – Addison-Wesley Publishing Company, New York.

PAPADIMITRIOU Christos H. [1997] – NP-Completeness: A Retrospective – *Lecture Notes in Computer Science* 1256, Springer, 1997 : 2-6.

PARTOUCHE Ariane [1998] – *Planification d'Horaires de Travail : Méthodologie, Modélisation et résolution à l'Aide de la Programmation Linéaire en Nombres Entiers et de la Programmation par Contraintes* – Thèse de l'Université Paris-Dauphine, Paris, France.
Disponible à : http://www.lamsade.dauphine.fr/Theses.html.

PEARSON Justin & JEAVONS Peter [1997] – A Survey of Tractable Constraint Satisfaction Problems – rapport technique CSD-TR-97-15, Royal Holloway, Université de Londres.
Disponible à : http://www.cs.rhul.ac.uk/research/constraints/publications/index.shtml.

PELLETIER Christine [1999] – *Application des Techniques d'Aide à la Décision à la Planification Sanitaire Régionale* – Thèse de l'Université Joseph Fourier, Grenoble, France.

PEREZ J. & BARBARA-ROMERO S. [1995] – Theory and Methodology. Three Practical Criteria of Comparison Among Ordinal Preference Aggregating Rules – *European Journal of Operational Research* 85, pp. 473-487.

PERETTI J. M. [1998] – *Ressources Humaines et Gestion de Personnel* – coll. Educapôle Gestion, 2e édition, Vuibert, avril 1998.

PERNY Patrice [1992] – Modélisation, Agrégation et Exploitation de Préférences Floues dans une Problématique de Rangement. Bases Axiomatiques, Procédures et Logiciels – Thèse de l'Université Paris-Dauphine, Paris, France, réimpr. Rapport de recherche Laforia 1993/09, Institut Blaise Pascal, Université Paris 6, Paris, France, mars 1993.

PERNY Patrice [1998] – Multicriteria Filtering Methods Based on Concordance and Non-Discordance Principles – *Annals of Operations Research* 80 : 137-165.

PERNY Patrice & SPANJAARD Olivier [2002*a*] – Modélisation des Préférences et Recherche dans les Graphes d'États – *2ndes Assises du Groupe de Travail I3*, 4-6 décembre 2002.
Disponible à : http://l1.lamsade.dauphine.fr/~spanjaar/publications.htm.

PERNY Patrice & SPANJAARD Olivier [2002*b*] – Preference-based Search in State Space Graphs – *Proceedings of the AAAI'2002 Conference*, Edmonton, Canada, 2002 : 751-756.

POMEROL Jean-Charles [1995] – Artificial Intelligence and Human Decision Making – Rapport de recherche Laforia/IBP 1995/12, Institut Blaise Pascal, Université Paris 6, Paris, France, mai 1995, 28 pages.

PUGET Jean-François [1998] – A Fast Algorithm for the Bound Consistency of Alldiff Constraints – *Proceedings of the Fifteenth National Conference on Artificial Intelligence* (*AAAI'98*), the MIT Press, juillet 1998 : 359-366.

PUGET Jean-François & LECONTE Michel [1995]– Beyond the Glass Box : Constraints as Objects – *Logic Programming : Proceedings of the th International Conference*, The MIT Press, Cambridge, Massachusetts, 513-527.

RAUZY Antoine [1995] – Polynomial Restrictions of SAT : What Can Be Done With an Efficient Implementation of the Davis and Putnam's Procedure – *Lecture Notes in Computer Science* 976, Springer-Verlag, New York : 515-532.

RAYNAUD Hervé [2000] – *L'Approche Axiomatique en Décision Multicritère – A paraître*.

RÉGIN Jean-Charles [1994] – A Filtering Algorithm for Constraints of Difference in CSPs – *American Association for Artificial Intelligence* (*AAAI'94*), Proceedings of the 12th National Conference on Artificial Intelligence, AAAI Press / MIT Press, Cambridge, 1994, 362-367.

RÉGIN Jean-Charles [1996] – Generalized Arc-Consistency for Global Cardinality Constraint – *American Association for Artificial Intelligence* (*AAAI'96*), Portland, 1996, 209-215.

RÉGIN Jean-Charles, PETIT Thierry, BESSIÈRE Christian, PUGET Jean-François [2001] – New Lower Bounds of Constraints Violations for Over-Constrained Problems – *in* Walsh T. (ed.): Principles and Practice of Constraint Programming. *Lecture Notes in Computer Science* 2239, Springer-Verlag, Berlin Heidelberg New York, 2001 : 332–345.

RÉGIN Jean-Charles & PUGET Jean-François [1997] – A Filtering Algorithm for Global Sequencing Constraints – *LNCS* 1330, *CP'97*, Springer-Verlag, Berlin, 1997, 32-46.

RÉGIN Jean-Charles & RUEHER Michel -[2000] – A Global Constraint Combining a Sum Constraint and Difference Constraints – *CP'2000*, *Lecture Notes in Computer Science* 1894, Springer-Verlag, Berlin, 2000 : 384-395.

REINHARDT Fritz & SOEDER Heinrich [1997] – *Atlas des Mathématiques* – coll. Encyclopédies d'Aujourd'hui, La Pochotèque, Le livre de Poche, Librairie Générale Française.

ROBERTS A. Wayne & VARBERG Dale E. [1973] – *Convex Functions* – coll. pure and applied mathematics, 1973.

ROBERTS Kevin W. S. [1980*a*] – Possibility Theorems with Interpersonally Comparable Welfare Levels – *Review of Economic Studies* 47 : 409-420.

ROBERTS Kevin W. S. [1980*b*] – Interpersonal Comparability and Social Choice Theory – *Review of Economic Studies* 47 : 421-439.

ROBERTS Kevin W.S. [1997] – The Structure of Interpersonal Comparisons: Discussion of Interpersonal Comparisons and the Possibility of Social Choice – *in* ARROW *et al*. [1997] : 230-236.

ROBERTSON Jack & WEBB William [1998] – *Cake-cutting Algorithms : Be Fair if You Can* – AK Peters Ltd., 1998.

ROSSI Francesca [1999] – Constraint (Logic) Programming: A Survey on Research and Applications – *Lecture Notes in Artificial Intelligence* 1865, Berlin, 1999 : 40-75.

ROWLEY Charles K. (ed.) [1993] – *Social choice theory* – coll. The international library of critical writings in economics ; 27 An Elgar reference collection, 3 volumes : vol. 1, The aggregation of preferences ; vol. 2, Utilitarian and contractarian goals ; vol. 3, Social justice and classical liberal goals, Aldershot, Brookfield.

ROY Bernard [1969] – Algèbre Moderne et Théorie des Graphes. Tome 1 : Notions et Résultats Fondamentaux – coll. Finance et Économie Appliquée, Dunod, Paris.

ROY Bernard [1970] – Algèbre Moderne et Théorie des Graphes. Tome 2 : Applications et Problèmes Spécifiques – coll. Finance et Économie Appliquée, Dunod, Paris.

ROY Bernard [1985] – *Méthodologie Multicritère d'Aide à la Décision* – Economica, Paris ; tr. angl. [1996] – *Multicriteria Methodology for Decision Aiding* – Kluwer Academic Publishers, Dordrecht.

ROY Bernard & BOUYSSOU Denis [1993] – *Aide Multicritère à la Décision : Méthodes et Cas* – Economica, coll. « Gestion », Paris.

RUSHMEIER Russell A., HOFFMAN Karla L. & PADBERG Manfred [1995] – *Recent Advances in Exact Optimization of Airline Scheduling Problems* – Rapport technique de l'Université George Mason, Virginie : 43 p..

RUSTEM Berc & VELUPILLAI Kumaraswamy [1990] – Rationality, Computability and Complexity – *Journal of Economic Dynamics and Control (Special Issue on Computer Science and Economics)* 14 (2), Mai 1990 : 419-432.

SAATY Thomas L. [1961] – *Elements of Queueing Theory with Applications* – McGraw-Hill, New York.

SAKAROVITCH Michel [1984*a*] – *Optimisation Combinatoire : Graphe et Programmation Linéaire* – Hermann, Paris.

SAKAROVITCH Michel [1984*b*] – *Optimisation Combinatoire : Programmation Discrète* – Hermann, Paris.

SAM-HAROUD Djamila & FALTINGS Boi [1996] – Consistency Techniques for Continuous Constraints – *Constraints, an International Journal* 1 (1-2), septembre 1996 : 85-118.

SARIN Subhash & AGGARWAL Sanjay [2001] – Modeling and Algorithmic Development of a Staff Scheduling Problem – *European Journal of Operational Research* 128 : 558-569.

SAVAGLIO Ernesto [2001] – *On Multidimensional Inequality : Ordering and Measurement* – Cahiers de recherche n°336, Département d'économie politique, Università degli Studi di Siena, Italie, décembre 2001 : 28 pages.

SAVAGLIO Ernesto [2002*a*] – *Multidimensional Inequality : A Survey* – Cahiers de recherche n°362, Département d'économie politique, Università degli Studi di Siena, Italie, aout 2002 : 19 pages.

SAVAGLIO Ernesto [2002*b*] – *On Multidimensional Inequality : A Majorisation Ordering and Its Representation* – Document de travail n° 2002-17, Groupe THEMA (Theorie Economique, Modélisation et Applications), Université Cergy-Pontoise & Université Paris X-Nanterre, octobre 2002 : 16 p..

SAVELSBERGH MartinW.P. [1994] – Preprocessing and Probing Techniques for Mixed Integer Programming Problems – *ORSA Journal on Computing* 6 (4) : 445-454.
Disponible à : http://udaloy.isye.gatech.edu/~mwps/.

SCHAEFER Thomas J. [1978] – The Complexity of Satisfiability Problems – *Proceedings of the 10th ACM Symposium on Theory of Computing* (STOC) : 216-226.

SCHÄRLIG Alain [1996] – *Electre et Prométhée. Un Complémentaire à Décider sur Plusieurs Critères* – presses polytechniques et universitaires romandes, Lausanne, Suisse.

SCHIEX Thomas, FARGIER Hélène & VERFAILLIE Gérard [1995] – Valued Constraint Satisfaction Problems: Hard and Easy Problems – *Proceedings of the IJCAI'95* , Morgan Kaufmann.

SCHRIJVER Alexander [1986] – *Theory of Linear and Integer Programming* – Series in Discrete Mathematics, Wiley-Interscience Publication, Chichester.

SCHRIJVER Alexander [2003] – *Combinatorial Optimization : Polyhedra and Efficiency* – Series in Algorithms and Combinatorics vol. 24, Springer Verlag Publication, Chichester.

SCHULZ Andreas S. [1996] – *Polytopes and Scheduling* – Thèse de l'Université Technique de Berlin, Berlin, Allemagne.

SEGAL M. [1974] – The Operator Scheduling Problem: A Network-Flow Approach – *Operations Research* 22: 808-823.

SEN Amartya Kumar [1970] – *Collective Choice and Social Welfare* – coll. Advanced textbooks in economics, vol. 11, Elsevier science publishers, Pays-Bas.

SEN Amartya Kumar [1971] – Choice Functions and Reavealed Preferences – *Review of Economic Studies* 38 (3), juillet 1971 : 307-317 ; réimpr. *in* SEN [1982*a*] : 41-53.

SEN Amartya Kumar [1972] – Interpersonal Aggregation and Partial Comparability : A Correction – *Econometrica* 40 (5), septembre 1972 : p. 959 ; réimpr. *in* SEN [1982*a*] : 203-221.

SEN Amartya Kumar [1973] – *On Economic Inequality* – Oxford University Press, Oxford, 1973 (1ère éd.), 1997 (édition élargie).

SEN Amartya Kumar [1977*a*] – On Weights and Measures : Informational Constraints in Social Welfare Analysis – *Econometrica* 45 : 1539-1572 ; réimpr. *in* SEN [1982*a*].

SEN Amartya Kumar [1977*b*] – Rational Fools : a Critique of the Behavioural Foundations of Economic Theory – *Philosophy and Public Affairs* 6 : 317-344 ; réimpr. *in* SEN [1982*a*] : 84-106 ; trad. fr. *in* SEN [1993*a*] : 87-116.

SEN Amartya Kumar [1980*a*] – Equality of What ? – *in* MAC MURRIN [1980] ; réimpr. *in* SEN [1982*a*] : 353-369 ; trad. fr. *in* SEN [1993*a*] : 189-213.

SEN Amartya Kumar [1980*b*] – Description as Choice – *Oxford Economic Papers* 32 (3) : 353-369 ; réimpr. *in* SEN [1982*a*].

SEN Amartya Kumar [1982*a*] – *Choice, Welfare and Measurement* – Oxford, Blackwell, and Cambridge, Massachusetts, MIT Press.

SEN Amartya Kumar [1982*b*] – Rights and Agency – *Philosophy and Public Affairs* 11 (1) : 3-39 ; trad. fr. *in* SEN [1993*a*] : 117-158.

SEN Amartya Kumar [1983] – Liberty and Social Choice – *Journal of Philosophy* 80 (1), janvier 1983 : 5-28 ; trad. fr. *in* Sen [1993*a*] : 159-188.

SEN Amartya Kumar [1986] – Social Choice Theory – chap. 22 *in* ARROW & INTRILIGATOR [1986] : 1073-1181.

SEN Amartya Kumar [1987] – *On Ethics and Economics* – Oxford, Blackwell publishers ; trad. fr. *in* SEN [1993] : 1-83.

SEN Amartya Kumar [1992] – *Inequality Reexamined* – Oxford University Press, Oxford, and Harvard University Press, Cambridge, MA ; tr. fr. – *Repenser l'Inégalité* – Editions du Seuil, mai 2000.

SEN Amartya Kumar [1993*a*] – *Éthique et Économie. Et Autres Essais* – coll. philosophie morale, Presses Universitaires de France, novembre 1993 (1re éd.), mars 1999 (2e éd.).

SEN Amartya Kumar [1993*b*] – Internal Consistency of Choice – *Econometrica* 61 (3) : 495-521.

SEN Amartya Kumar [1996] – Social Commitment and Democracy: The Demands of Equity and Financial Conservatism –*in* BARKER [1996] ; tr. fr. – Responsabilité Sociale et Démocratie : L'Impératif d'Équité et le Conservatisme Financier – *in* SEN [1999*a*] : 77-124.

SEN Amartya Kumar [1997*a*] – Maximization of the Act of Choice – *Econometrica* 65 (4), juillet 1997 : 745-779.

SEN Amartya Kumar [1997*b*] – Individual Preference as the Basis of Social Choice – *in* ARROW *et al*. [1997], vol. 1 : 15-37.

SEN Amartya Kumar [1999*a*] – *L'Économie est une Science Morale* – Editions La Découverte, Paris.

SEN Amartya Kumar [1999*b*] – The Possibility of Social Choice – *American Economic Review* 89 (3), juin 1999 : 349-378.

SIFERD Sue Perrott & BENTON W. C. [1992] – Workforce Staffing and Scheduling : Hospital Nursing Specific Models – *European Journal of Operational Research* 60 : 233-246.

SIMON Herbert A. [1977] – *The New Science of Management Decision* – édition révisée, Englewood Cliffs : Prentice-Hall, 1977.

SIRGY Joseph M., EFRATY David, SIEGEL Phillip & LEE Dong-Jin [2001] – A New Measure of Quality of Work Life (QWL) Based on Need Satisfaction and Spillover Theories – *Social Indicators Research* 55 : 241-302.

SLANY Wolfgang [1994] – *Fuzzy Scheduling* – rapport technique 94/66 du Christian Doppler Laboratory for Expert Systems, et de l'Université Technique de Vienne, Autriche.
Disponible à : http://www.dbai.tuwien.ac.at:8080/staff/slang.html.

SOLNON Christine [1997*a*] – Coopération de Solveurs Linéaires sur les Réels pour la Résolution de Problèmes Linéaires sur les Entiers – *Journées Francophones de Programmation Logique par Contraintes* (*JFPLC'97*), 1997 : 211-225.

SOLNON Christine [1997*b*] – Cooperation of LP Solvers for Solving MILPs – *International Conference on Tools for Artificial Intelligence* (*ICTAI*), IEEE Computer Society Press, 1997 : 240-247.

SU Francis Edouard [2000] – Book Review : Cake-Cutting Algorithms by J. Robertson and W. Webb — *American Mathematical Monthly* 107 : 185-188.

SUZUMURA Kotaro [1983] – *Rational Choice, Collective Decisions and Social Welfare* – Cambridge University Press, Cambridge.

SUZUMURA Kotaro [1997] – Interpersonal Comparisons and Justice – *in* ARROW *et al*. [1997] : 202-229.

SUZUMURA Kotaro [2002] – Introduction – *in* ARROW *et al*. [2002].

THEVENET Maurice [1993] – L'Implication au Travail – *Sciences Humaines* 28, mai 1993.

TEGHEM Jacques [1996] – *Programmation Linéaire* – coll. Statistique & Mathématiques Appliquées, Ellipses, 1996, Paris.

THIENEL Stefan [1995] – *ABACUS - A Branch-And-CUt System* – Thèse de l'Université de Cologne (Köln), Cologne, Allemagne, 1995.
Disponible à : http://www.informatik.uni-koeln.de/ls_juenger/staff/thienel/publications.html.

THOMPSON Gary M. [1995] – Improved Implicit Optimal Modeling of the Labor Shift Scheduling Problem – *Management Science* 41 (4): 595- 607.

THOMSON William [1997*a*] – On the Axiomatic Method. Part I : A User's Guide – *Rochester Center for Economic Research Working Paper* 445, novembre 1997.

THOMSON William [1997*b*] – On the Axiomatic Method. Part II : Its Scope and Recent Applications to Game Theory and Resource Allocation – *Rochester Center for Economic Research Working Paper* 446, novembre 1997.

TORRENS Marc & FALTINGS Boi [2002] – Using Soft CSPs for Approximating Pareto Optimal Solutions Sets – *Proceedings of the AAAI Workshop on Preferences in Constraint Satisfaction*.

TROTTER William T. Jr. [1992] – *Combinatorics and Partially Ordered Sets. Dimension Theory* – Johns Hopkins University Press, Baltimore, London.

TSOUKIÀS Alexis [1999] – *Sur la Généralisation des Concepts de Concordance et de Discordance en Aide Multicritère à la Décision* – Document du LAMSADE n°117, Université Paris-Dauphine, novembre 1999.

TSUI Kai-yuen [1999] – Multidimensional Inequality and Multidimensional Generalized Entropy Measures: An Axiomatic – *Social Choice and Welfare* 16 : 145-157.

ULUNGU E. L. & TEGHEM Jacques [1994] – Multi-Objective Combinatorial Optimization Problems: A Survey – *Journal of Multi-Criteria Decision Analysis* 3, pp. 83-104.

VAN BEEK Peter & DECHTER Rina [1995] – On the Minimality and Global Consistency of Row-Convex Constraint Networks – *Journal of the ACM*, 42(3), 1995 : 543-561.

VAN HENTENRYCK Pascal [1989] – *Constraint satisfaction in logic programming* – coll. Logic programming series, MIT Press, London ; Cambridge (Mass), 224 p.

VAN HENTENRYCK Pascal & DEVILLE Yves [1991] – The Cardinality Operator : A New Logical Connective for Constraint Logic Programming – *Logic Programming : Proceedings of the 8th International Conference* (*ICLP'91*), MIT Press, Cambridge, Massachusetts 1991 : 745-759.

VANSNICK J. C. [1986] – *De Borda et Condorcet à l'Agrégation Multicritère* – cahiers du Lamsade n° 70, Université Paris-Dauphine, Paris, mai 1986.

VERFAILLIE Gérard [1997] – *Texte de Synthèse des Travaux* – Thèse d'Habilitation à Diriger les Recherches, Université Paul Sabatier, Toulouse, France, 28 octobre 1997.
Disponible à : http://www.cert.fr/francais/deri/verfaillie/newoc/node5.html.

VERFAILLIE Gérard & LOBJOIS Lionel [1999] – Problèmes Incohérents : Expliquer l'Incohérence, Restaurer la Cohérence – *in Actes des 5ièmes Journées Nationales sur la Résolution Pratique de Problèmes NP-Complets* (*JNPC'99*), Lyon, France, 1999 : 111-120.

VERFAILLIE Gérard, MARTINEZ David & BESSIÈRE Christian [1999] – A Generic Customizable Framework for Inverse Local Consistency – *in Proceedings of the 16th National Conference on Artificial Intelligence and 11th Conference on Innovative Applications of Artificial Intelligence AAAI\IAAI'99*, 18-22 juillet 1999, Orlando, Floride, USA, AAAI Press / The MIT Press : 169-174.

VERFAILLIE Gérard & SCHIEX Thomas [1995] – Maintien de Solution dans les Problèmes Dynamiques de Satisfaction de Contraintes : Bilan de Quelques Approches – *Revue d'Intelligence Artificielle* 9 (3) : 269-309.

VINCKE Philippe [1977] – Quasi-Kernels of Minimum Weakness in a Graph – *Discrete Mathematics* 20 (2) :187-192.

VINCKE Philippe [1989] – *L'Aide Multicritère à la Décision* – coll. Statistique et Mathématiques Appliquées, éd. Ellipses (France) et éd. De l'Université de Bruxelles (Belgique) ; tr. angl. et mise-à-jour [1992] – *Multicriteria Decision-Aid* – John Wiley & Sons.

VON NEUMANN John & MORGENSTERN Oskar [1944] – *Theory of Games and Economic Behavior* – Princeton university press, Princeton.

WARNER D. Michaël [1976] – Scheduling Nursing Personnel According to Nursing Preference : A Mathematical Programming Approach – *Operations Research* 24 (5), septembre-octobre 1976 : 842-856.

WHITE D. J. [1977] – Kernels of Preferences Structures – *Econometrica* 45(1), janvier 1977 : 91-100.

WEIL Georges, HEUS Kamel, FRANÇOIS Patrice [1994] – Gymnaste : Aide à l'élaboration des roulements infirmiers. Du traitement des absences au management participatif – *in* DEMONGEOT *et al*. [1994].

WEIL Georges & KORNBLUM Claude [1994] – Les difficultés culturelles d'introduction de l'informatique à l'hôpital – *in* DEMONGEOT *et al*. [1994].

WIJNGAARD Jacob [1983] – Aggregation in Manpower Planning – *Management Science* 29 (12), décembre 1983 : 1427-1435.

WOODRUFF D. L. (ed.) [1998] – *Advances in Computational and Stochastic Optimization, Logic Programming, and Heuristic Search* – Kluwer Academic Publishers, Boston, MA, 1998.

ZHANG Weixiong [1996] – *Branch-and-Bound Search Algorithms and their Computational Complexity* – rapport de recherche n°ISI-RR-96-44318, Université de Caroline du Sud, Etats-Unis, mai 1996 : 31 pages.

ZHENG Buhong [2001] – Poverty Orderings: A Graphical Illustration – *Social Choice and Welfare* 18, Springer-Verlag : 165-178.

ZHOU Jianyang [1996] – A Constraint Program for Solving the Job-shop Problem – *Lecture Notes in Computer Science* 1118, Springer-Verlag, Berlin, 1996 : 510-524.

ZHOU Jianyang [1997] – A Permutation-based Approach for Solving the Job-shop Problem – *Constraints : An International Journal* 2 (2), octobre 1997 : 185-213.

ZIEGLER Gunter M. [1995] – *Lectures on Polytopes* – coll. Graduate Texts in Mathematics 152, Springer-Verlag, New-York, 1995.

Index des Noms

A

Abdennadher, S. 44, 61, 293
Aggarwal, S. 308
Aichholzer, O. 126, 293
Aït Younes, A. 117, 137, 293
Andjiga, N. G. 96, 97, 98, 304
Arrow, K. J.76, 136, 139, 272, 293, 295, 297, 304, 307, 309, 310
Aspremont d', C. 103, 127, 130, 166, 193, 293
Aykin, T. 66, 67, 293
Azibi, R. 117, 137, 293

B

Bafman, R. I. 43, 44, 117, 148, 295
Baker, K. R. 62, 293
Balas, E. 35, 293
Bana e Costa, C. A. 85, 86, 91, 120, 293
Bang-Jensen, J. 32, 294
Barbara-Romero, S. 305
Barker, P. 294, 309
Barnett, W. A. 77, 294
Basu, K. 129, 178, 294
Beaumont, N. 66, 294
Bechtold, S. E. 66, 294
Beldiceanu, N. 42, 62, 263, 294
Belmandt, Z. 96, 97, 98, 283, 294
Benhamou, F. 38, 39, 222, 224, 294
Bentham, J. 76, 137
Benton, W. C. 22, 47, 310
Berge, C. 32, 108, 109, 111, 112, 175, 294
Berthod, V. 23, 49, 300
Bessière, C.44, 87, 90, 155, 223, 224, 277, 280, 281, 294, 306, 311
Billera, L. J. 174, 175, 220, 295
Billionet, A. 61, 66, 295
Birkhoff, G. 174, 249
Bistarelli, S. 44, 157, 295
Blackorby, C. 193, 295
Blazewicz, J. 70
Bleuzen-Guernalec, N. 238, 240, 295
Boissonnat, J. D. 220, 295
Boizumault, P. 43, 90, 302
Bonmati, J. M. 22, 64, 295
Borda, J.-C. 76, 136, 311
Bossert, W. 193, 295
Bourguignon, F. 186, 295
Boutilier, C. 43, 44, 117, 148, 295
Bouyssou, D.72, 76, 78, 83, 86, 87, 91, 95, 96, 101, 103, 105, 106, 109, 117, 121, 124, 125, 126, 127, 128, 129, 131, 135, 136, 137, 144, 146, 149, 198, 209, 218, 295, 303, 307
Brandstädt, A. 112, 296
Brissaud, M. 96, 97, 98, 304
Brusco, M. J. 65, 66, 296
Buchanan, J. 140, 297
Bulatov, A. A. 42, 296

C

Cai, X. 66, 73, 296
Carr, R. 247, 249, 296
Caseau, Y. 61, 62, 267, 296
Chan, P.23, 24, 44, 51, 61, 62, 63, 84, 147, 267, 296, 301
Charnes, A. 62, 296
Chartrand, G. 126, 296
Cheng, B. M. W. 59, 60, 61, 296
Choi, K. M. F. 60, 61, 296
Cohen, D. A. 42, 301
Colmerauer, A. 238, 240, 295
Condorcet, M. J.-A. C. M. 76, 136, 311
Cooper, M. C. 87, 277, 297
Cooper, W. W. 62, 296
Corner, J. 140, 297
Cosytec 36, 42, 297
Coté, M. J. 47, 297
Courbon, J.-C. 78, 85, 89, 297
Crète, J. 27, 297

D

Dago, P. 91, 297
Dahl, G. 180, 183, 297
Dalton, H. 168
Dantzig, G. B. 23, 64, 297
Dasgupta, P. 175, 176, 297
Dechter, R. 87, 297, 311
Demongeot, J. 297, 312
Denicoló, V. 144, 145, 297
Deville, Y. 41, 156, 223, 263, 311
Didrit, O. 38, 222, 224, 235, 236, 301
Donaldson, D. 193, 295
Dubois, D. 78, 85, 89, 297
Durand, S. 78, 91, 120, 121, 136, 298

E

Ecker, K. 70
Edmonds, J. 247
Efraty, D. 201, 310
Ehrgott, M. 137, 170, 215, 298
Eisenbrand, F. 180, 298
Eiter, T. 32, 298
Elster, J. 130
Emelichev, V. A. 170, 298
Équipe MUSTIL 79, 272, 298

F

Fages, F. 40, 209, 298
Faltings, B. 39, 211, 214, 236, 298, 308, 311
Fargier, H. 44, 157, 295, 308

Ferber, J. 140, 298
Ferland, J. A. 54, 55, 61, 62, 298, 299
Fishburn, P. C. 105, 141, 298
Floudas, C. A. 35, 299
Fodor, J. 100, 299
Foster, J. E. 75, 116, 177, 185, 186, 187, 193, 198, 205, 221, 299
François, P. 22, 24, 44, 312
Freuder, E. C... 36, 87, 223, 277, 280, 294, 299, 303
Frost, D. H. 209, 265, 299
Fujito, T. 247, 249, 296

G

Gandibleux, X. 137, 170, 215, 298
Garbay, C. 77, 299
Garey, M. R. 27, 31, 32, 38, 79, 299
Gascon, V. 61, 62, 299
Gauvin, A. 23, 204, 299
Gavanelli, M. 149, 157, 209, 210, 211, 213, 214, 225, 226, 299
Geib, C. H. 43, 44, 117, 148, 295
Gevers, L. 103, 127, 130, 166, 193, 293
Ghoshal, J. 109, 110, 299
Gondran, M. 147, 180, 300
Gottlob, G. 32, 298
Greco, S. 119, 141, 300
Greenberg, H. J. 91, 300
Greene, C. 295
Guillo, P. Y. 61, 62, 267, 296
Guitouni, A. 139, 300
Gutin, G. 32, 294
Gyssens, M. 42, 301

H

Hammond, P. J. 130, 143, 192, 300
Hansen, P. 69, 300
Hardy, G. 174, 175, 185, 220, 221
Haynes, T. 109, 111, 120, 299, 300
Hedetniemi, S. 109, 111, 120, 299, 300
Henderson, T. C. 280, 304
Henig, M. 140, 297
Henning, M. A. 109, 300
Hertz, A. 23, 49, 54, 298, 300
Heus, K. 21, 22, 24, 44, 61, 62, 300, 312
Hickey, T. J. 38, 39, 224, 235, 300
Hiroux, M. 147, 301
Hoffman, K .L. 22, 23, 66, 68, 72, 307
Hoos, H. H. 43, 44, 117, 148, 295
Hung, S. L. 70, 303
Hylland, A. 130

I

Ilog 36, 42, 44, 62, 301
Imbeau, L. M. 27, 297
Intriligator, M. D. 293, 309

J

Jacobs, L. W. 66, 294, 296
Jacot, H. 23, 204, 299
Jacquet-Lagrèze, E. 66, 301
Jaszkiewicz, A. . 61, 62, 73, 141, 165, 202, 267, 301
Jaulin, L. 38, 222, 224, 235, 236, 301
Jaumard, B. 24, 54, 55, 69, 72, 130, 165, 202, 300, 301
Jeavons, P. 42, 296, 301, 305
Johns, T. R. 65, 66, 296
Johnson, D. S. 27, 31, 32, 38, 79, 299, 301
Johnson, E. 247
Joseph, R. R. 147, 260, 301
Ju, Q. 38, 39, 224, 235, 301
Jussien, N. 43, 90, 302

K

Keeney, R. L. 76, 140, 155, 302
Kieffer, M. 38, 222, 224, 235, 236, 301
Kobeissi, M. 126, 302
Kodratoff, Y. 77
Köhler, G. 136, 302
Konjevod, G. 247, 249, 296
Kornblum, C. 312
Koshevoy, G. 181, 183, 302
Kostreva, M. M. 149, 215, 220, 302
Krisher, R. 63, 64, 302
Krivine, J.-L. 302
Krokhin, A. A. 42, 296

L

Lafage, C. 43, 44, 117, 302
Lahiri, S. 145, 302
Lai, K. F. 70, 303
Lang, J. 43, 44, 117, 147, 302
Laporte, G. 63, 302
Larrosa, J. 223, 294
Laskar, R. 109, 110, 299
Lavoie, A. 54, 55, 298
Le Beux, P. 297, 312
Le, V. B. 112, 296
Leconte, M. 40, 306
Lee, D.-J. 201, 310
Lee, J. H. M. 59, 60, 61, 296
Lesniak, L. 126, 296
Levenez, E. 61, 62, 267, 296
Lévine, P... 78, 79, 81, 82, 83, 84, 85, 121, 170, 302
Li, K. N. 66, 73, 296
Li, L. X. 66, 303
Li, N. 66, 303
Lin, C. K. Y. 70, 303
List, C. 181, 303
Littlewood, J. 174, 175, 185, 220, 221
Lobjois, L. 84, 89, 158, 311
Lorenz, M. O. 173

M

Maasoumi, E. 181, 186, 303
mac Murrin, S. M. 303, 309
Mackworth, A. K. 36, 279, 303
Mandler, M. 188, 303
Marathe, M. V. 42, 303
Marchant, T. 95, 127, 135, 136, 144, 295, 303
Marcotte, O. 146, 303

Marquardsen, P. 37, 305
Marshall, A. W. 168, 176, 185, 303
Martel, J.-M. 139, 300
Martinez, D. 87, 88, 224, 277, 280, 303, 311
Marziliano, P. 54, 298
Masini, G. 280, 304
Matarazo, B. 119, 141, 300
Mehlhorn, K. 238, 240, 303
Messeguer, P. 223, 294
Meyer auf'm Hofe, H. 61, 303
Michelon, P. 61, 62, 299
Mietus, D. M. 25, 61, 63, 304
Miller, H. 54, 55, 56, 304
Minoux, M. 147, 180, 232, 300, 304
Mintzberg, H. 22, 78, 304
Mohr, R. 280, 304
Montanari, U. 36, 44, 157, 295, 304
Montaut, D. 66, 301
Moondra, S. L. 66, 304
Morgenstern, O. 109, 312
Morin, J.-M. 22, 304
Moulen, J. 96, 97, 98, 304
Moulin, H. 77, 134, 197, 294, 304
Mousseau, V. 76, 128, 304
Musliu, N. 61, 304

N

Narasimhan, R. 66, 70, 305
Nemhauser, G. L. 34, 35, 209, 305
Newell, A. 83, 305
Nielsen, S. R. 37, 305

O

Ogryczak, W. 149, 215, 219, 220, 221, 302, 305
Ok, E. A. 129, 177, 178, 299, 305
Oldenkamp, J. H. 25, 61, 62, 63, 72, 305
Older, W. J. 38, 39, 222, 224, 294
Olkin, I. 168, 176, 185, 303
Othmani, I. 72, 83, 128, 134, 305

P

Padberg, M. 22, 23, 66, 68, 72, 305, 307
Papadimitriou, C. H.27, 30, 32, 43, 79, 278, 301, 305
Parekh, O. 247, 249, 296
Pareto, V. 137
Partouche, A.21, 23, 45, 47, 48, 50, 51, 61, 62, 65, 66, 68, 70, 73, 170, 267, 301, 305
Pearson, J. 42, 305
Pelletier, C. 271, 305
Perepelitsa, V. A. 170, 298
Peretti, J. M. 22, 306
Perez, J. 305
Perny, P.79, 81, 95, 96, 99, 100, 105, 124, 125, 135, 136, 141, 144, 145, 214, 295, 303, 306
Pesch, E. 70
Petit, T. 44, 90, 155, 224, 306
Pierskalla, W. P. 54, 55, 56, 304
Pigou, A. C. 168
Pillone, D. 109, 110, 299
Pirlot, M. 95, 103, 135, 136, 144, 295, 303
Pisinger, D. 37, 305
Polyà, G. 174, 175, 185, 220, 221
Pomerol, J.-C.78, 79, 81, 82, 83, 84, 85, 89, 121, 170, 297, 302, 306
Poole, D. 43, 44, 117, 148, 295
Prade, H. 85, 297
Puget, J.-F. 40, 44, 90, 155, 224, 263, 276, 306

R

Raïffa, H. 76, 140, 155, 302
Rath, G. J. 54, 55, 56, 304
Rauzy, A. 306
Raynaud, H. 71, 120, 136, 139, 141, 144, 293, 306
Régin, J.-C.44, 62, 90, 155, 224, 235, 239, 262, 263, 276, 294, 306, 307
Reinhardt, F. 126, 180, 232, 307
Rinaldi, G. 180, 298
Robert, V. 23, 49, 300
Roberts, A. W. 175, 307
Roberts, K. W. S. 129, 144, 307
Robertson, J. 197, 307, 310
Rossi, F. 36, 44, 157, 295, 307
Roubens, M. 100, 299
Rowley, C. K. 76, 307
Roy, B.71, 76, 78, 80, 83, 85, 86, 87, 88, 89, 91, 96, 101, 102, 103, 105, 106, 108, 109, 111, 117, 121, 124, 125, 126, 128, 129, 131, 136, 137, 140, 146, 149, 198, 209, 218, 293, 297, 307
Rueher, M. 155, 224, 235, 239, 307
Rushmeier, R. A. 22, 23, 66, 68, 72, 307

S

Saaty, T. L. 47, 307
Sakarovitch, M.27, 32, 33, 71, 88, 91, 147, 169, 209, 247, 284, 308
Salles, M. 77, 294
Sam-Haroud, D. 39, 236, 308
Sarangarajan, A. 174, 175, 220, 295
Sarin, S. 308
Savaglio, E. 181, 308
Savelsbergh, M. W. P. 282, 308
Schaefer, T. J. 42, 308
Schärlig, A. 136, 137, 308
Schiex, T. 44, 90, 157, 295, 308, 312
Schlenker, H. 44, 61, 293
Schmidt, G. 70
Schofield, N. J. 77, 294
Schrijver, A. 34, 38, 180, 247, 249, 308
Schulz, A. S. 175, 308
Segal, M. 24, 47, 66, 308
Semet, F. 24, 54, 55, 69, 72, 130, 165, 202, 301
Sen, A. K.75, 76, 92, 103, 104, 105, 106, 107, 111, 116, 117, 127, 129, 130, 131, 136, 138, 139, 143, 144, 145, 160, 161, 166, 168, 171, 172, 173, 175, 176, 177, 179, 180, 185, 186, 187, 188, 189, 191, 192, 193, 194, 195, 196, 197, 198, 199, 214, 221, 272, 287, 293, 295, 297, 299, 304, 307, 308, 310
Siegel, P. 201, 310

Siferd, S. P. 22, 47, 310
Simion, R. 295
Simon, H. A. 77, 83, 84, 92, 272, 273, 305, 310
Sirgy, J. M. 201, 310
Slany, W. 310
Slater, P. J. 109, 111, 120, 299, 300
Slowinski, R. 119, 141, 300
Soeder, H. 126, 180, 232, 307
Soland, R. M. 146, 303
Solnon, C. 282, 310
Spanjaard, O. 145, 214, 306
Spinrad, J. P. 112, 296
Stanley, R. P. 295
Starrett, D. 175, 176, 297
Stein, W. E. 47, 297
Su, F. E. 197, 310
Suppes, P. 189
Suzumura, K. 76, 104, 105, 111, 129, 130, 131, 132, 154, 160, 161, 162, 163, 171, 192, 196, 197, 293, 295, 304, 307, 310
Swinkels, G.M. 240

T

Teghem, J. 146, 170, 310, 311
Thevenet, M. 64, 310
Thiel, S. 238, 240, 303
Thienel, S. 209, 310
Thompson, G. M. 66, 310
Thomson, W. 134, 141, 142, 197, 304, 311
Torrens, M. 211, 214, 311
Trotter, W. T. Jr. 153, 213, 311
Tsoukiàs, A. 95, 100, 105, 135, 136, 144, 272, 295, 303, 311
Tsui, K. 181, 183, 186, 311

U

Ulungu, E. L. 170, 311

V

van Beek, P. 87, 297, 311
van Emden, M. H. 38, 39, 224, 235, 240, 301
van Hentenryck, P. 40, 41, 156, 223, 263, 311
Vansnick, J. C. 311
Varberg, D. E. 175, 307
Ventura, P. 180, 298
Verfaillie, G. 44, 84, 85, 87, 88, 89, 90, 91, 157, 158, 224, 277, 280, 295, 297, 303, 308, 311
Villeneuve, S. 61, 62, 299
Vincke, P. 72, 76, 85, 95, 100, 103, 110, 126, 128, 135, 136, 137, 139, 144, 146, 198, 295, 300, 303, 312
von Neumann, J. 109, 174, 249, 312
Vovor, T. 24, 54, 55, 69, 72, 130, 165, 202, 300, 301

W

Wallace, R. J. 299, 303
Walter, O. 38, 222, 224, 235, 236, 301
Warner, D. M. ... 24, 47, 53, 54, 57, 58, 72, 165, 312
Webb, W. 197, 307, 310
Weglarz, J. 70
Weil, G. 22, 24, 44, 63, 147, 296, 297, 301, 312
White, D. J. 312
Wijngaard, J. 69, 70, 312
Wolsey, L. A. 34, 35, 209, 305
Woodruff, D. L. 300, 312
Wu, J. C. K. 59, 60, 61, 296

Y

Yannakakis, M. 32, 301
Yvinec, M. 220, 295

Z

Zhang, W. 209, 312
Zheng, B. 167, 312
Zhou, J. 240, 312
Ziegler, G. M. 34, 174, 220, 312

Index Terminologique

A

absorption **109**
acyclicité
  dans les graphes 69, 140
  des plannings 21
  P- — ..... *Voir* P-acyclicité
agis **129**
algorithme **28**
  adaptatif 32
ALLDIFF 42, 233, 234, 240, **276**, 282, 303, 306
arité **37**
assignment-type problem 54
attitude fondamentale **96**
auto-planification 25, **63**, 72, 88, 89, 139, 197
axiome
  de choix **86**
  de consistance **87**
  de Lorenz corrélation croissante (CIL) **182**
  de mesurabilité **104**, 132, 139
  de Nash **163**
  de rnp-stabilité **162**
  d'égalité des proportions (EP)**168**, 170, 172, 177, 181, 197, 216, 289
  d'identité **131**, 142, 195, 225, 243
  d'identité complète **132**, 194
  d'invariance **104**, **132**
  d'invariance sous les permutations (IP)**168**, 171, 177, 181, 184, 189, 216, 289
  d'invariance sous les réplications (PRI)179, 183, 203
  independance of irrelevant alternatives 143

B

Borda
  fonction de choix de — **136**
  règle de — **136**, 138, 143, 144
  règle de — généralisée **136**

C

ceteris paribus 117, 200, 295
charge **47**
coalition 123, 124, 170, 171
  concordante **124**, 136, 137, 288
  discordante **125**, 137
cocircuit **284**, 286
concerné ..... *Voir* observé
Condorcet
  vainqueur de — **136**, 144
conflit
  global **133**, 145, 166
  local **124**, 134
connexité **33**, 284
  forte — **33**, 114, 120, 284
consistance
  arc-— (AC)231, 261, 262, 277, **278**, 280, 294, 298, 306
  borne — 88, 235, 238, 239, 241, 242, 276
  globale inverse — (GIC)**87**, 222, 225, 227, 229, 233, 235, 252, 276
  qualitative 91, **161**, 163, 213
contrainte **37**
  dure (impérative) ..... **43** *Voir* contrainte
  globale **42**
  souple (molle, flexible) **43**
  souple collective **154**
  souple de propriété 43
  souple finale 157
  souple individuelle **154**
couplage ..18, 33, 174, 180, 209, 240, 249, 255, 270
courbe
  de charge 47
  de Lorenz**173**, 176, 180, 183, 185, 203, 219, 232, 238
couverture
  de la charge 63, 264
  des sommets d'un graphe18, 209, 244, 250, 252, 270
critère
  d'intervalle **102**, 103, 104, 188
  fonction- — ... **101**, 104, 126, 132, 137, 147, 188
  pseudo- — **103**
  quasi- — 103
  vrai- — **28**, **102**
cyclicité
  de la charge 47, 70
  des plannings 21, 49, 51, 61, 62, 73, 293, 296

D

days-off scheduling problem **51**
dominance **139**
  de Lorenz **173**
  de Lorenz Généralisée **184**
  de Pareto **139**
domination (graphe orienté)106, **109**, 111, 116, 299, 300

E

e.r.m.d.m.i. **116**, 118
e.r.m.m.i.**112**, 113, 114, 115, 116, 160, 171, 285, 286
e.r.o.m.i.**112**, 113, 114, 115, 116, 118, 160, 171, 284, 285
ELECTRE 136, 141
émetteur **129**
ensemble de choix **105**
  collectif **133**
ensemble relativement maximal
  dominant minimal pour l'inclusion*Voir* e.r.m.d.m.i.

minimal pour l'inclusion ..... *Voir* e.r.m.m.i.
ensemble relativement optimal
minimal pour l'inclusion ..... *Voir* e.r.o.m.i.
étiquette **46**
extension
d'une règle d'agrégation139, 156, 176, 177, 183, 184, 187, 189, 191, 192, 194, 213, 215, 217, 221, 269, 270, 289
d'une relation binaire.....**93**

F

fair cake division problem **166**
fermeture transitive.....**33**, 106, 111, 184, 185
fonction
de choix.....**105**, 145, 160
de préférence.....**98**, 117, 134, 135, 143
objective..... *Voir* vrai-critère
forward-checking.....262, 277, **280**, 294

G

GCC.....42, 233, 234, 262, **276**
graduation.....101, **104**
graphe17, 27, 32, 125, 209, 269, 271, 294, 300, 302, 307
biparti..... 174, 180, 240, 244, 254, 270, 283, 291
d'états.....214, 306
d'agrégation.....140, 157, 194, 199, 201
de solution.....**246**, 292
orienté **32**, 69, 93, 94, 106, 149, 203, 244, 284

H

heuristique 214
méta- — 141, 214, 267, 271
recherche —.....83
hiérarchie agrégative**140**, 144, 157, 194, 199, 201, 227, 252, 269

I

indépendance(graphe orienté)..... *Voir* stable
indiscernabilité..... 106, 117, 145, 151, 286, 287
individu.....**123**
interactivité ..... *Voir* SIAD
intervalle **38**
généralisé **38**
ordre d'— **95**, 102, 175
intervenant **129**

J

jeux
théorie des —..... 77, 109, 111, 165, 188, 197

K

knapsack problem 70

L

lexicographique
règle — **137**, 143, 170, 202
leximin.....189, **192**, 193

M

majorisation 177, **183**
majorité
règle de la —95, **136**, 138, 139, 142, 170, 171, 195
markotope.....**183**, 221
matrice
bistochastique **174**, 175, 181, 220, **249**
de permutation **174**, 175, 181, 295
d'incidence 147, **246**, 249
doublement stochastique.....**174**
ligne-stochastique **183**
maximalité **106**
relative 106, **112**, 116, 171
maximin.....189, **191**
mesure relative d'inégalité **186**
minimax.....**191**
modèle
de faisabilité.....**28**
préférentiel.....**28**
multipermutoèdre **175**, 176, 183, 215, 220, 221, 242
généralisé **220**
multiple shift scheduling problem.....**51**

N

noyau d'un graphe106, **109**, 111, 116, 120, 137, 141, 171, 283

O

observé.....**129**
opérateur de cardinalité **41**, 156, 245, 250, 263
optimalité.....**106**
relative 106, **112**, 116, 171

P

P-acyclicité**95**, 97, 102, 107, 138, 153, 157, 161, 162, 209, 215
paradoxe
de Sen 195
du choisi-rejeté.....**120**
Pareto
règle de — **137**
PB-contrainte souple
étendue.....**151**
simple.....**149**
simple sur les positions.....**150**
période **45**
permutoèdre.....**174**, 175
polyèdre convexe.....**34**
polytope **34**
d'affectation **174**, 249
des ordres totaux **175**
position (*station*).....**129**
préférences
binaires.....**96**
collectives **135**
conditionnelles.....44, 148
entre écart de préférences **101**, 103
individuelles.....**123**
managériales **201**

principe de transfert
de Foster-Sen (FST) **193**
de Pigou-Dalton (PDT) **169**
problème
combinatoire **28**
d'aide à la décision **82**
d'appartenance **30**, 39, 79, 86, 149, 278
de comparaison **80**, 100, 149, 211
de consistance simple **30**, 31, 42
de construction de tours (PCT) **50**
de décision **28**
de dénombrement **29**
de positionnement des repos (PPR) **51**
de pure répartition **166**
de pure répartition multidimensionnelle (PRM) **180**
de satisfaction de contraintes (CSP) **37**
de vérification de l'optimalité **80**
de voisinage **149**
d'élaboration de grilles étiquetées (PEGE) **51**
d'énumération **29**
d'évaluation **80**
partiel **30**, 82
partiellement automatisé **82**
sous- — **28**
technique ..... *Voir* problème d'aide à la décision
profil
étendu **130**
généralisé **130**
pondéré **128**, 132, 137
simple **123**
type sympathie étendue **129**

Q

qualité
de service15, 18, 23, 24, 47, 54, 58, 62, 63, 64, 89, **201**, 260, 270
de vie15, 24, 63, **201**, 202, 204, 206, 260, 270, 310

R

real-full-lookhead 262, 266, 277, **280**
recouvrement
des sommets d'un graphe ..... *Voir* couverture
récursivité 40, 41, 84, 126, 148, 156, 204, 269
réduction polynomiale **31**, 38
règle d'agrégation **134**
étendue **134**
partielle 139
simple **134**
règle de choix
collectif **134**
collectif simple **134**
collectif type sympathie étendue **134**
règle de décision **86**
collective **135**
règle de rangement interpersonnelle (IROR) **138**, 144, 194
relation binaire **92**
de préférence valuée **98**
floue 99
relation non-binaire de préférence **105**
réseau de contraintes .... *Voir* système de contraintes

S

Schur-concavité **175**, 176, 184, 221
self-scheduling ..... *Voir* auto-planification
sémantique de contrainte **37**
SIAD **81**
single-shift scheduling problem **51**
solution
choisie **105**
de meilleur compromis **133**
d'un graphe ..... *Voir* noyau
équitable **165**
équitable et efficace **165**
maximale **106**
optimale **106**
satisfaisante 82
sortie-polynomialité en temps 32, 115, 236
stable (graphe orienté) **108**, 109, 111, 114, 115, 118, 120, 158, 162, 209, 213, 214, 283
Suppes
indifférence de **189**
principe de **189**
Suppes-Sen
contrainte souple de — **244**
règle d'équité de — (SSE) **189**
règle d'équité de — étendue (ESSE) **191**
système de contraintes **36**
basé sur les préférences (P-CS) **157**
système multi-agent 141, 298

T

théorie des jeux ..... *Voir* jeux
transformation polynomiale
de contraintes **42**
de problèmes de décision **31**
transitivité
inter-solution — **190**, **195**, **252**
quasi- — **95**, 139, 157, 162, 209, 214

U

utilité 70, 80, 188
additive **137**
individuelle 166

V

vacation **46**
type de — **46**
VC-contrainte souple
étendue **152**
simple **44**
vecteur d'incidence **246**, 249, 291

Z

zonotope de Lorenz **183**

Titre. Systèmes Interactifs d'Aide à l'Élaboration de Plannings de Travail de Personnel : Contraintes, Aide à la Décision, Représentation Combinatoire des Préférences, Équité et Résolution par Décomposition Arborescente et par Consistance.

Résumé. L'élaboration de plannings de travail de personnel représente un problème majeur auquel sont confrontées la plupart des organisations. Nous nous préoccupons ici de l'automatisation de cette tâche, liant négociation et calcul combinatoire.
Les modèles préférentiels sur lesquels se base l'évaluation des plannings sont complètement révisés. Une nouvelle représentation combinatoire des préférences est proposée, couplant le concept de relation binaire de préférence à celui de contrainte souple. Elle est mieux adaptée à la formulation des opinions de chaque acteur et intègre la notion d'aide à la décision et d'interactivité en admettant l'incomparabilité et l'intransitivité. Ensuite, l'agrégation des préférences individuelles est réalisée par la méthode axiomatique en usant de plusieurs principes éthiques : l'équité (Pigou-Dalton, Suppes-Sen) et l'efficacité (Pareto).
Divers algorithmes de filtrage et de recherche de solution adaptés à la forme particulière des préférences et critères sont conçus. Un système interactif d'élaboration de plannings infirmiers est décrit.

Mots clés. Programmation par contraintes, contraintes souples, contraintes globales, aide à la décision, SIAD, équité, théorie du choix social, élaboration de plannings, planification.

Title. Personnel Timetables Design Support Systems: Constraints, Decision Support, Combinatorial Preference Elicitation, Equity and Solving by way of Branch and Bound Procedures and Consistency.

Abstract. Personnel timetabling is a major problem which concerns most of the organizations. We are focusing here on the automatisation of this task combining heavy computation and bargaining between the different actors (workers, managers and customers).

A complete review of usual timetabling preference models is performed. We propose a new model, preference constraint, combining preference binary relation from decision theory and soft constraint from constraint programming. By admitting incomparability and intransitivity, this model is more fitted to actors' preference elicitation, interaction and decision support. New methods for individual preference aggregation are introduced including ethical principles: equity (Pigou-Dalton, Suppes-Sen) and efficiency (Pareto). These principles are used to define generic compromise criteria as quality of work life induced by schedules.

Next, filtering and search algorithms have been designed for some preference constraints dedicated to previously defined criteria. Finally a nurse scheduling decision support system is described.

Keywords. Constraint programming, soft constraints, global constraints, decision support systems, social choice theory, inequality measurement, employee timetabling, scheduling.