



**HAL**  
open science

# Les matroïdes et leur implication dans l'allocation de ressources indivisibles : algorithmes d'approximation avec garantie de performance

Lydia Tlilane

► **To cite this version:**

Lydia Tlilane. Les matroïdes et leur implication dans l'allocation de ressources indivisibles : algorithmes d'approximation avec garantie de performance. Autre [cs.OH]. Université Paris Dauphine - Paris IX, 2014. Français. NNT : 2014PA090068 . tel-01134377

**HAL Id: tel-01134377**

**<https://theses.hal.science/tel-01134377>**

Submitted on 23 Mar 2015

**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.

École Doctorale de Dauphine  
Laboratoire d'Analyse et Modélisation de Systèmes pour  
l'Aide à la DÉcision (LAMSADE)

---

# THÈSE

en vue de l'obtention du diplôme de

**Docteur en Informatique**

de l'Université Paris-Dauphine

---

## **Les matroïdes et leur implication dans l'allocation de ressources indivisibles : algorithmes d'approximation avec garantie de performance**

---

Présentée par

Lydia TLILANE

Soutenue publiquement le 28 novembre 2014

### **Jury :**

<i>Directeur :</i>	Jérôme MONNOT	CNRS, Université Paris-Dauphine
<i>Rapporteurs :</i>	Christian LAFOREST	Université Blaise Pascal
	Patrice PERNY	Université Pierre et Marie Curie
<i>Examineurs :</i>	Sylvain BOUVERET	Ensimag, Grenoble-INP
	Laurent GOURVÈS	CNRS, Université Paris-Dauphine
	Jérôme LANG	CNRS, Université Paris-Dauphine
	Pierre MARQUIS	Université d'Artois



---

# *Remerciements*

En préambule à cette thèse, je souhaite exprimer ma gratitude à tous ceux qui ont contribué à son bon déroulement.

En premier lieu, j'exprime ma vive reconnaissance à Jérôme et Laurent qui m'ont offert l'opportunité de faire ma recherche à leur côté et m'ont fait l'honneur d'accepter de diriger ma thèse. Leurs idées fécondes, leur rigueur scientifique, leurs relectures pointilleuses ont largement contribué à l'accomplissement de ce travail. J'ai énormément appris avec eux scientifiquement, et j'ai particulièrement apprécié l'orientation que nous avons prise en se rapprochant de la communauté d'intelligence artificielle et du choix social computationnel. Je suis également très admirative de leurs qualités humaines. Jérôme, Laurent, je vous remercie infiniment pour votre investissement, pour tout le temps que vous m'avez patiemment consacré, pour vos orientations et pour vos encouragements perpétuels.

Mes sincères remerciements vont aussi à Christian Laforest et Patrice Perny pour l'intérêt qu'ils ont porté à mon travail en acceptant d'être rapporteurs de ma thèse malgré leurs charges professionnelles et pour leur travail minutieux de relecture. J'exprime également ma profonde gratitude à Jérôme Lang pour avoir présidé mon jury de soutenance et auprès de qui j'ai beaucoup appris sur le choix social computationnel. Mes remerciements vont également à Sylvain Bouveret pour avoir examiné mon travail et pour ses remarques très pointues sur le partage équitable de biens indivisibles. Je remercie aussi Pierre Marquis pour avoir accepté d'examiner mon travail et pour ses remarques pertinentes, notamment sur l'oracle d'indépendance et l'input de nos algorithmes. À tous les membres du jury, merci d'avoir lu et critiqué mon travail et merci pour toutes vos remarques constructives et vos suggestions qui ont contribué à améliorer la qualité de ce manuscrit.

Je n'oublierai pas d'exprimer mes vifs remerciements à tous les membres du projet ANR GUEPARD 2010-2013 (Guaranteed Efficiency for Pareto Optimal Solutions Determination) qui ont contribué à ma formation en optimisation combinatoire multi-critère et pour avoir financé la majorité de mes missions durant ma thèse. J'ai beaucoup apprécié les échanges que nous avons eus durant les réunions de travail et en conférences. Merci à Mohamed Ali Aloulou, Cristina Bazgan, Lyes Belhoul, Liangliang Fu, Lucie Galand, Florian Jamain, Renaud Lacour, Dalal Madakat et Daniel Vanderpooten du LAMSADE; Audrey Cerqueus, Xavier Gandibleux, Anthony Przybylski et Thomas Vincent du LINA; Anisse Ismaili, Julien Lesca, Thibault Lust, Patrice Perny, Olivier Spanjaard et Paul Weng du LIP6.

---

Je dois des égards aux chercheurs en intelligence artificielle qui m'ont aidé à intégrer leur communauté. Mes remerciements vont à Jérôme Lang, Nicolas Maudet et Olivier Spanjaard pour l'intérêt qu'ils ont porté à mon travail et pour leurs précieux conseils et orientations. Je remercie aussi Souhila Arib et Abdallah Saffidine pour leurs conseils et encouragements et pour m'avoir informé des conférences et autres groupes de travail en IA. Mes remerciements vont également aux membres des projets ANR COCA (Combinatorial Optimization with Competing Agents) et LARDONS (Learning And Reasoning for Deciding Optimally using Numerical and Symbolic information) pour avoir financé certaines de mes missions.

Plusieurs personnes m'ont aidé dans la relecture de mes articles et/ou de ma thèse. Merci à Amal Benhamiche, Hakan Kutucu, Sébastien Martin et Raouia Taktak. Ma reconnaissance va également à Eirlys Da-costa, enseignante d'Anglais à Dauphine, qui m'a beaucoup aidé à perfectionner mon anglais et qui a corrigé minutieusement un article.

À tous les membres du LAMSADE, j'ai beaucoup apprécié nos collaborations, qu'elles soient scientifiques, pédagogiques, administratives, ou rien que pour le plaisir de vous avoir croisé et avoir échangé quelques mots dans la salle café du LAMSADE. Merci à Alexis Tsoukias, directeur du LAMSADE, qui m'a offert à quelques reprises des ouvrages de littérature choisis par ses soins. À l'équipe administrative, Katerina, Mireille, Nathalie et Olivier, merci pour votre disponibilité, votre efficacité et votre joie de vivre communicative. Corinne, ton insatiable bonne humeur a rendu ma thèse et nos réunions de travail très agréables. Sans oublier notre petit groupe de natation, Amine, Edouard, Emeric, Olivier et Renaud, j'ai beaucoup apprécié d'évoluer sportivement avec vous.

À tous les doctorants que j'ai eu le plaisir de connaître durant ma thèse : Amal, Dalal, Irène et Souhila, votre présence, votre soutien et vos conseils furent très précieux et nos nombreuses discussions m'ont permis d'apprécier plus la vie au LAMSADE. Mes remerciements vont aussi à Edouard, Emeric (pour nos échanges passionnés sur le féminisme et sur Tikz!), Florian, Lyes, Nico, Renaud (l'éternel chef des thésards) et Sébastien (pour tout); et tous les autres, Amine, Anaëlle, Diana, Ian-Christopher, Liangliang, Maria-Laura, Marek, Morgan, Myriam, Mohamed, Mohammed-Amine, Nathanaël, Pierre-Emmanuel, Raja, Raouia, Raphaël, Romain, Sami, Satya, Sonia, Thomas, Tom, Yann, Yassine, Youcef ... sans oublier Sofia et Viviana du CEREMADE. De l'autre côté des murs de Dauphine, je n'oublie pas Hanane du LIPN qui m'a accompagné depuis le début de mes études en recherche opérationnelle, Kamel de l'INRIA et Anisse du LIP6. Merci pour tous ces moments que nous avons partagé.

À titre privé, j'adresse mes plus sincères remerciements à mes parents et à tous mes proches, qui m'ont toujours soutenu et encouragé. À toutes et à tous, merci.

# Table des matières

<b>Remerciements</b>	<b>i</b>
<b>Introduction</b>	<b>1</b>
Contexte . . . . .	1
Contribution . . . . .	4
<b>I État de l’art</b>	<b>9</b>
<b>1 Théories de l’approximation et des matroïdes</b>	<b>11</b>
1.1 Théorie de l’approximation en optimisation mono-objectif . . . . .	11
1.1.1 Généralités sur l’approximation . . . . .	12
1.1.2 Conception d’algorithmes d’approximation . . . . .	14
1.2 Approximation en optimisation multi-objectif . . . . .	15
1.2.1 Approche budget ( $\epsilon$ – <i>constraint</i> ) . . . . .	17
1.2.2 Approximation de l’ensemble de Pareto . . . . .	18
1.2.3 Approximation du point idéal . . . . .	21
1.2.4 Inapproximabilité en optimisation multi-objectif . . . . .	23
1.3 Théorie des matroïdes . . . . .	24
1.3.1 Généralités sur les matroïdes . . . . .	24
1.3.2 Exemples de matroïdes . . . . .	27
1.3.3 Matroïdes pondérés . . . . .	31
1.3.4 Matroïdes et approximation en multi-objectif . . . . .	40
<b>2 Problèmes d’allocation de ressources et notions d’équité</b>	<b>43</b>
2.1 Problème <i>générique</i> d’allocation de ressources . . . . .	43
2.1.1 Problème <i>classique</i> d’allocation de ressources . . . . .	48
2.1.2 Généralisation du problème classique aux matroïdes . . . . .	49
2.2 Notions de bien-être social et d’équité . . . . .	54
2.3 Garantie dans le pire cas . . . . .	62
2.3.1 Les fonctions $V_n$ . . . . .	63
2.3.2 Une propriété clé sur $V_n$ . . . . .	65
2.4 Algorithmes d’allocation . . . . .	66
2.4.1 Algorithmes décentralisés et protocoles de partage . . . . .	67

2.4.2	Algorithmes à véracité garantie . . . . .	71
<b>II</b>	<b>Contributions</b>	<b>73</b>
<b>3</b>	<b>Équité à un élément près</b>	<b>75</b>
	Introduction . . . . .	76
3.1	Le modèle . . . . .	78
3.2	Équitabilité . . . . .	78
3.3	Absence de jalousie . . . . .	81
3.4	Proportionnalité . . . . .	83
3.5	Absence d'envie . . . . .	88
	3.5.1 Proche de l'équilibre . . . . .	93
	3.5.2 Proche de l'absence d'envie pour deux agents . . . . .	93
	Conclusion et perspectives . . . . .	95
<b>4</b>	<b>Garantie dans le pire cas : une approche centralisée</b>	<b>97</b>
	Introduction . . . . .	98
4.1	Le modèle . . . . .	100
	4.1.1 Les fonctions $W_n$ . . . . .	101
4.2	L'algorithme SEUIL . . . . .	103
4.3	Une légère modification de SEUIL pour deux agents . . . . .	112
4.4	Borne supérieure sur l'utilité de l'agent le moins heureux . . . . .	114
	4.4.1 Conclusion sur la valeur de $t_n$ . . . . .	119
	Conclusion et perspectives . . . . .	120
<b>5</b>	<b>Garantie dans le pire cas : une approche décentralisée</b>	<b>123</b>
	Introduction . . . . .	124
5.1	Le modèle . . . . .	125
5.2	Un protocole pour deux agents . . . . .	126
5.3	Un protocole pour trois agents . . . . .	128
5.4	Un protocole général pour $n \leq 9$ agents . . . . .	129
	5.4.1 Le point de vue du diviseur . . . . .	133
	5.4.2 Les points de vue des non-diviseurs pour $n \geq 3$ agents . . . . .	134
	5.4.3 Preuve du théorème principal . . . . .	135
	Conclusion et perspectives . . . . .	138
<b>6</b>	<b>Solutions de compromis pour des matroïdes pondérés coloriés</b>	<b>141</b>
	Introduction . . . . .	142
6.1	Contexte . . . . .	144
	6.1.1 Le modèle . . . . .	144

6.1.2	Approximation non triviale . . . . .	145
6.2	Un algorithme général glouton . . . . .	146
6.3	Utilités additives . . . . .	152
6.4	Complexité générale et une solution particulière . . . . .	155
6.5	Cas des couleurs non-pondérées . . . . .	156
6.5.1	Un algorithme à 3-phases . . . . .	157
6.5.2	Base de poids maximum avec un nombre exact de couleurs . . . . .	161
	Conclusion et perspectives . . . . .	164
<b>Conclusion et perspectives</b>		<b>167</b>
	Conclusion . . . . .	167
	Perspectives . . . . .	169
<b>III Annexes</b>		<b>173</b>
<b>A Preuve du lemme 2.2</b>		<b>175</b>
<b>B Preuves des propriétés 4.1, 4.2 et 4.3</b>		<b>179</b>
B.1	Preuve de la propriété 4.1 . . . . .	179
B.2	Preuve de la propriété 4.2 . . . . .	180
B.3	Preuve de la propriété 4.3 . . . . .	180
<b>C La garantie de SEUIL est au plus près</b>		<b>181</b>
<b>D Preuves des lemmes 5.1 et 5.2</b>		<b>187</b>
D.1	Preuve du lemme 5.1 . . . . .	187
D.2	Preuve du lemme 5.2 . . . . .	188
<b>E Abréviations et notations</b>		<b>193</b>
E.1	Abréviations . . . . .	193
E.2	Notations . . . . .	193
<b>F Liste des problèmes abordés</b>		<b>197</b>
<b>G Liste des illustrations</b>		<b>201</b>
G.1	Algorithmes et protocoles . . . . .	201
G.2	Figures . . . . .	201
G.3	Tableaux . . . . .	202
<b>Bibliographie</b>		<b>202</b>
<b>Résumé</b>		<b>217</b>





# Introduction

## Contexte

La décision collective est une problématique essentielle qui se retrouve dans toutes les strates de la société. Elle permet d'élire les représentants d'un gouvernement, choisir une date de réunion qui arrange tous les participants, établir un programme de vacances approuvé par toute la famille, choisir un groupe syndicaliste représentant le personnel d'une société, etc. Nous nous intéressons dans cette thèse à la décision collective. L'objectif est de prendre une décision consensuelle parmi un ensemble d'alternatives soumises au point de vue d'une collectivité.

La théorie de la décision collective a des aspects pluridisciplinaires. Par exemple, la psychologie sociale se préoccupe de l'interaction sociale dans la collectivité. L'économie, de par la théorie du choix social, s'intéresse au concept de décision consensuelle. Les mathématiques permettent de formaliser cette problématique, alors que l'informatique s'intéresse à l'aspect *computationnel* de la décision.

La décision collective, dans le domaine de l'informatique, s'inscrit dans le cadre de plusieurs communautés. En intelligence artificielle, les systèmes multi-agents permettent de modéliser les processus de décision collective. Ils définissent aussi les règles régissant les interactions entre les membres de la collectivité. En aide à la décision, le but est de rechercher un consensus au sein de la collectivité. Ce consensus peut être établi au moyen d'une fonction d'agrégation sur les préférences des membres de la collectivité. En informatique théorique, on s'intéresse à la complexité d'obtention d'une solution consensuelle et de sa construction algorithmique. Toutes ces thématiques se joignent pour former la théorie du choix social computationnel qui traite, analyse et construit des méthodes dédiées à la décision collective (Chevalyere *et al.*, 2007b).

Notre approche du problème se place au carrefour de l'optimisation combinatoire et du choix social computationnel. Ainsi, les problèmes que nous traitons font appel à une modélisation par une structure combinatoire classique : les matroïdes. Nous utilisons les outils de l'informatique théorique tels que la théorie de la complexité, l'algorithmique et l'approximation avec garantie de performance. Enfin, la problématique traitée et les concepts appliqués dans cette thèse proviennent de la théorie du choix social computationnel.

Le but de cette thèse est de déterminer une solution de compromis pour des problèmes soumis à de multiples points de vue. Nous développons cette problématique à travers les concepts suivants : solution, points de vue et compromis.

La *solution* est déterminée parmi plusieurs alternatives possibles, c'est-à-dire qu'elle doit être réalisable pour le problème considéré. Nous nous intéressons dans le cadre de cette thèse à des problèmes qui ont une structure combinatoire. Plus précisément, il s'agit de la classe des systèmes d'ensembles qui ont une structure de matroïde. Un matroïde est défini par un ensemble fini  $X$  d'éléments et une collection  $\mathcal{F}$  de sous-ensembles de  $X$  qui satisfont les propriétés suivantes : (i) l'ensemble vide appartient à la collection  $\mathcal{F}$  ; (ii) l'hérédité qui signifie que si un ensemble  $F_i \in \mathcal{F}$  alors tout sous-ensemble  $F_j \subseteq F_i$  appartient également à  $\mathcal{F}$  ; (iii) la propriété d'augmentation qui signifie que pour tout couple  $F_1, F_2 \in \mathcal{F}$  tel que  $|F_1| < |F_2|$ , il existe  $x \in F_2 \setminus F_1$  tel que  $F_1 \cup \{x\}$  est aussi dans la collection  $\mathcal{F}$ . Les ensembles qui appartiennent à  $\mathcal{F}$  sont dits indépendants et les ensembles indépendants maximaux pour l'inclusion sont les bases du matroïde  $\mathcal{M}$ . La théorie des matroïdes est centrale en optimisation combinatoire, elle a permis d'unifier des structures apparemment séparées comme les arbres et les transversaux dans les graphes et elle a engendré des algorithmes efficaces pour résoudre des problèmes d'optimisation non triviaux en temps polynomial. Le point commun entre tous les problèmes traités dans cette thèse est qu'une solution réalisable est toujours définie par une base d'un matroïde. Nous reviendrons plus en détail sur la définition des matroïdes et des bases dans la section 1.3.

Les matroïdes ont des applications diverses et permettent de modéliser des problèmes connus dans plusieurs disciplines. En algèbre linéaire, le matroïde matriciel modélise l'ensemble des colonnes linéairement indépendantes d'une matrice. En ordonnancement, le matroïde transversal permet de capturer les ordonnancements réalisables de tâches unitaires sur une machine. Dans les graphes, le matroïde graphique permet de modéliser les forêts et les arbres couvrants. Le matroïde de couplage permet de caractériser les sommets saturés par un couplage dans un graphe. Le matroïde des chemins disjoints caractérise les couples de sommets admettant des chemins arêtes disjointes dans un graphe. En choix social, le matroïde uniforme permet de modéliser des problèmes d'élection à vainqueurs multiples. Le matroïde laminaire permet de sélectionner des représentants de communautés hiérarchiques. Enfin, le matroïde de partition modélise des problèmes d'affectation et d'allocation de ressources. Puisque nous travaillons sur un matroïde général, nos résultats s'appliquent à tous ces problèmes.

Nous portons un intérêt particulier au problème *classique* d'allocation de biens indivisibles qui se modélise à l'aide du matroïde de partition. Le problème classique d'allocation de biens indivisibles consiste à répartir, de façon équitable, un

ensemble d'objets indivisibles sur un ensemble d'agents. C'est un problème usuel qui s'applique par exemple lors d'un héritage ou encore dans le partage de biens en cas de divorce. Le but étant de satisfaire les parties concernées en tenant compte de leurs préférences représentées sous forme d'utilités pour l'ensemble des biens. C'est un thème central en choix social computationnel (Chevaleyre *et al.*, 2006).

La collectivité est constituée d'un groupe de membres qui expriment leurs *points de vue* sur toutes les alternatives (ou solutions réalisables). Les membres de la collectivité ainsi que leurs points de vue peuvent être formalisés à l'aide de l'un des concepts suivants :

- **Agents** : les agents définissent les membres de la collectivité. Chaque agent exprime son point de vue sur les alternatives (ou solutions réalisables) à travers des préférences ordinales ou cardinales (fonctions d'utilité). Il s'agit alors de l'étude d'un système multi-agents. Le principe d'équité d'une solution est ici crucial.
- **Objectifs (ou critères)** : Le point de vue de chaque membre s'exprime via une fonction objectif associée d'une direction d'optimisation (maximisation ou minimisation). La problématique de décision collective correspond ici à un problème d'optimisation multi-objectif.
- **Scénarios** : chaque membre de la collectivité est assimilé à un scénario qui peut se réaliser. La réalisation d'un scénario induit une satisfaction sur chaque acte (ou solution réalisable). L'objectif est de rechercher une solution robuste, c'est-à-dire qui est satisfaisante à l'égard de n'importe quel scénario qui pourrait se réaliser. Ce formalisme caractérise la décision dans l'incertain.

Bien que ces trois aspects aient des interprétations différentes, leur formalisme reste analogue pour les problèmes que nous traitons. Nous utiliserons dans cette thèse la terminologie des systèmes multi-agents. Nous ferons également appel à certains concepts en optimisation multi-objectif pour caractériser la solution de compromis.

Un *compromis* est une solution réalisable qui satisfait au mieux tous les membres de la collectivité (les agents, les objectifs ou les scénarios). La solution de compromis doit également être équitable pour tous les membres. Il convient alors de définir la notion d'équité.

L'*équité* d'une solution est une caractéristique importante que toute solution de compromis se doit de respecter. L'équité constitue une part importante de la théorie du choix social (Steinhaus, 1948; Young, 1995; Brams et Taylor, 1996; Moulin, 2004). Young (1995) soutient que l'équité ne se résume pas à une formule et qu'il n'existe pas de principe d'équité qui soit objectif. L'équité doit tenir compte

de la satisfaction de la collectivité, appelée aussi bien-être social (*social welfare*) en théorie du choix social. À cette fin, plusieurs concepts de bien-être social ont été définis dans la littérature. Par exemple, le critère utilitariste qui cherche la plus grande satisfaction globale en maximisant la somme des utilités des agents pour la solution de compromis. Un autre exemple est le critère égalitariste qui cherche à rendre le plus heureux possible, l'agent le plus défavorisé. Ces principes d'équité peuvent être appliqués pour choisir une solution de compromis.

Notre attention porte sur l'aspect computationnel de l'obtention d'une solution de compromis. En particulier, nous nous intéressons à des constructions algorithmiques de cette solution. La solution de compromis peut être déterminée par les agents en interagissant ensemble, il s'agit alors de déterminer un algorithme ou protocole dit décentralisé. Il est également possible qu'un coordinateur prenne la décision finale, dans ce cas la quête d'un algorithme centralisé est plus adéquate.

Puisque nous généralisons le problème classique d'allocation de biens indivisibles, il est possible qu'il n'existe pas d'allocation qui satisfasse les critères d'équité les plus courants (absence d'envie, etc.). De plus, de façon générale il est  $\text{NP}$ -complet de décider de l'existence d'une telle solution. Ce résultat est valable pour le problème général de la base d'un matroïde multi-agents (ou multi-objectif). Par ailleurs, il est notoire que la plupart des problèmes d'optimisation combinatoire mono-objectifs deviennent  $\text{NP}$ -complets dès lors qu'un objectif supplémentaire est ajouté. Ce constat est malheureusement valable pour déterminer l'existence d'une solution (base) d'un matroïde multi-objectif qui satisfait certaines performances. Une alternative pour trouver une solution de compromis est de construire une solution approchée avec garantie de performance en temps polynomial.

La théorie de l'approximation s'intéresse à fournir des solutions approchées pour des problèmes d'optimisation avec garantie de performance dans le pire des cas. Nous nous intéressons à fournir des algorithmes d'approximation polynomiaux pour déterminer une solution de compromis pour des problèmes multi-agents (ou multi-objectifs). Cela permet de s'intéresser à la qualité de la solution de compromis en fournissant une garantie par rapport à l'optimum. Nous nous concentrons sur l'approximation du point idéal du problème multi-objectif. Le point idéal est une solution rarement réalisable dont l'image atteint l'optimum sur tous les objectifs. Cela permet de donner une garantie sur l'utilité de chaque agent pour la solution de compromis par rapport à son optimum.

## Contribution

Nous nous intéressons à des problèmes dont les solutions sont définies par un matroïde. Une solution réalisable est une base évaluée par plusieurs agents

au moyen de fonctions d'utilité. Chaque agent souhaite maximiser son utilité. L'objectif est de trouver une solution de compromis et de la partager sur les agents, en tenant compte de l'équité du partage de cette solution.

Ce manuscrit est organisé comme suit :

Le chapitre 1 présente le cadre général des théories abordées et des techniques appliquées. La section 1.1 présente la théorie de l'approximation polynomiale pour des problèmes d'optimisation mono-objectifs. La section 1.2 présente les concepts les plus répandus pour approcher des problèmes multi-objectifs avec garantie de performance. La section 1.3 définit la théorie des matroïdes en ciblant certains concepts qui seront utilisés dans le développement des prochains chapitres. Dans cette même section, nous apportons notre contribution à la théorie des matroïdes, en particulier sur le partitionnement des bases et leur interchangeabilité (section 1.3.3).

Le chapitre 2 retrace les problèmes d'allocation de ressources. La section 2.1 introduit un problème générique d'allocation de ressources et présente quelques cas particuliers comme le problème *classique* d'allocation de biens indivisibles, et sa généralisation aux matroïdes qui est le problème que nous étudions dans cette thèse. Le problème classique d'allocation de biens indivisibles revêt d'une importance particulière dans ce manuscrit puisqu'il représente un cas particulier de notre problème et pour lequel nous améliorons quelques résultats d'approximabilité. Certains concepts d'équité seront également définis dans la section 2.2 afin de les appliquer à notre problème.

Le problème traité dans cette thèse est défini par un matroïde. Chaque élément a  $n$  utilités représentant les points de vue de  $n \geq 2$  agents. Ces utilités sont toutes additives, c'est-à-dire que l'utilité d'un agent pour un sous-ensemble d'éléments est la somme de ses utilités pour les éléments de ce sous-ensemble. L'objectif est de trouver une base et de la partitionner en  $n$  parts réparties sur les agents. Ce problème généralise le problème classique d'allocation de biens indivisibles.

Dans le chapitre 3, nous nous intéressons à l'équité du partage d'une base d'un matroïde sur  $n$  agents. Nous étudions quelques notions d'équité qui sont couramment utilisées dans les problèmes d'allocation de ressources. Il s'agit de l'équitabilité (section 3.2), l'absence de jalousie que nous introduisons (section 3.3), la proportionnalité (section 3.4) et l'absence d'envie (section 3.5). Nous adaptons les définitions de ces notions au partage d'une base d'un matroïde. Nous proposons également des relaxations de ces concepts afin de garantir l'existence d'une solution équitable selon ces définitions relaxées, et sa construction en temps polynomial. Cette relaxation consiste à atteindre la notion d'équité considérée à un élément près, et sera dite proche de l'équité.

Dans les chapitres 4 et 5, nous nous intéressons à fournir une base d'un matroïde partitionnée en  $n$  parts, avec une garantie dans le pire cas sur l'utilité de chaque agent pour sa part. Les fonctions d'utilités des agents sont toutes additives. D'un point de vue multi-objectif, le problème consiste à trouver une solution réalisable qui est une approximation du point idéal du problème avec une garantie de performance. De façon générale, ce problème n'admet pas de rapport d'approximation constant même pour  $n = 2$ . Nous nous intéressons alors à fournir des résultats d'approximation en fonction de  $n$  et d'un paramètre, noté  $\alpha_i$  ( $i = 1, \dots, n$ ), représentant l'utilité maximum de l'agent  $i$  pour un élément du matroïde dans une instance normalisée. Cette approche étend les travaux de Demko et Hill (1988); Markakis et Psomas (2011) établis pour le problème classique d'allocation de biens indivisibles.

Dans le chapitre 4, nous proposons un algorithme polynomial centralisé qui construit une base partitionnée en  $n$  parts avec une garantie sur l'utilité de chaque agent  $i$  ( $i = 1, \dots, n$ ) en fonction de  $n$  et de  $\alpha_i$  (section 4.2). Cette garantie est au moins aussi bonne que celle connue pour le problème classique d'allocation de biens indivisibles et est valable pour tous les problèmes ayant une structure de matroïde. Une meilleure garantie est proposée dans la section 4.3 pour le cas de  $n = 2$  agents. Nous fournissons également une borne supérieure sur l'utilité de l'agent le plus défavorisé pour tout  $n \geq 2$  dans la section 4.4.

Dans le chapitre 5, nous proposons, pour le même problème, un algorithme décentralisé qui fournit une garantie, toujours en fonction du paramètre  $\alpha_i$  ( $i = 1, \dots, n$ ) pour  $n = 2$  (section 5.2),  $n = 3$  (section 5.3) et  $n \leq 9$  (section 5.4) agents. L'intérêt des algorithmes décentralisés est de permettre aux agents d'interagir librement afin qu'ils construisent ensemble une solution. Ces algorithmes permettent également de réduire la communication de certaines informations que les agents peuvent juger privées comme leurs utilités. La garantie obtenue améliore la garantie existante pour le problème classique d'allocation de biens indivisibles.

Le chapitre 6 étudie une extension du problème précédent (dans le cas de deux agents) à un type particulier d'utilités sous-modulaires. L'objectif est de proposer quelques pistes de recherche lorsqu'on modélise les utilités des agents par des fonctions plus complexes. Chaque élément du matroïde est évalué par deux agents différents qui ont des fonctions d'utilités hétérogènes. La première fonction d'utilité est additive. La seconde fonction d'utilité est une coloration pondérée, c'est-à-dire que chaque élément du matroïde a une couleur et chaque couleur a un coût. Le coût des couleurs d'une solution est la somme des coûts de chaque couleur. Même si une couleur apparaît plusieurs fois, on ne compte son coût qu'une seule fois. Cette fonction d'utilité est sous-modulaire et est connue sous le nom de prix chromatique (*the chromatic price*) dans sa version minimisation (Maffioli *et al.*, 2007). Nous proposons des algorithmes polynomiaux qui construisent une solution

de compromis avec garantie de performance pour plusieurs variantes du problème. Les garanties proposées sont des approximations du point idéal et sont toutes à rapport constant.

Tous les algorithmes présentés dans ce manuscrit sont déterministes et polynomiaux, sous l'hypothèse que l'oracle qui détermine si un sous-ensemble d'éléments est indépendant pour un matroïde donné est polynomial. Tous les résultats (théorèmes, propositions, etc.) non originaux sont suivis d'une référence.

Nous concluons ce rapport par quelques perspectives de recherche autour de la problématique traitée.





**Première partie**

**État de l'art**



# Théories de l'approximation et des matroïdes

Ce chapitre présente le cadre général dans lequel s'inscrit cette thèse, ainsi que les outils nécessaires pour le développement des chapitres suivants. À savoir, les algorithmes d'approximation en optimisation mono-objectif et multi-objectif et la théorie des matroïdes.

## Sommaire

---

<b>Contexte</b> . . . . .	<b>1</b>
<b>Contribution</b> . . . . .	<b>4</b>

---

## 1.1 Théorie de l'approximation en optimisation mono-objectif

Nous considérons que les bases de la théorie de la complexité sont connues du lecteur. Autrement, nous invitons les personnes intéressées à lire Garey et Johnson (1979). Nous supposons dans ce qui suit que  $\mathbb{P} \neq \text{NP}$ .

En optimisation combinatoire, beaucoup de problèmes d'optimisation sont intraitables, c'est-à-dire qu'il est peu probable de les résoudre à l'optimum en temps polynomial. Une solution naturelle est d'utiliser des méthodes approchées qui fournissent certaines garanties sur la qualité de la solution en temps polynomial.

Les algorithmes d'approximation avec garantie de performance sont des méthodes non exactes qui produisent des solutions réalisables en temps polynomial avec des bornes sur leur qualité. Ils sont apparus dans les années 1960 (Vizing, 1964; Graham, 1966) avant la construction de la théorie de la  $\text{NP}$ -complétude qui a été formalisée par Cook (1971). Une définition formelle des algorithmes d'approximation a été proposée par Garey *et al.* (1972).

Nous présentons dans cette section quelques concepts généraux sur les algorithmes d'approximation pour des problèmes d'optimisation mono-objectifs.

### 1.1.1 Généralités sur l'approximation

Soit  $I$  une instance d'un problème d'optimisation  $\pi$ . Notons  $X_I$  l'ensemble des solutions réalisables de  $I$ . Si  $x \in X_I$  alors  $VAL(I, x)$  est la valeur de cette solution. Soit  $OPT(I)$  la valeur optimale de  $I$ . De façon formelle, les solutions approchées sont définies comme suit.

**Définition 1.1 (Solution approchée)** *Une solution réalisable  $x \in X_I$  est  $\epsilon$ -approchée pour une instance  $I$  de  $\pi$  avec  $\epsilon \in ]0, 1[$  pour un problème de maximisation (ou  $\epsilon > 1$  pour un problème de minimisation), si sa valeur  $VAL(I, x)$  est garantie d'être à un facteur  $\epsilon$  de l'optimum  $OPT(I)$ , c'est-à-dire  $\frac{VAL(I, x)}{OPT(I)} \geq \epsilon$  pour un problème de maximisation (ou  $\frac{VAL(I, x)}{OPT(I)} \leq \epsilon$  pour un problème de minimisation).*

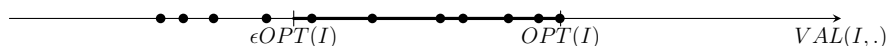


FIGURE 1.1 – Solutions  $\epsilon$ -approchées pour un problème de maximisation.

L'axe représenté dans la figure 1.1 contient des points représentant les valeurs des solutions réalisables pour une instance  $I$  d'un problème de maximisation. Les solutions dont les valeurs se trouvent entre  $\epsilon OPT(I)$  et  $OPT(I)$  (segment de droite en gras) sont des  $\epsilon$ -approximations du problème où  $\epsilon \in ]0, 1[$ .

**Définition 1.2 (Algorithme d'approximation)** *Un algorithme  $\mathcal{A}$  est dit  $\epsilon$ -approché si pour toute instance  $I$  de  $\pi$ ,  $\mathcal{A}$  retourne une solution  $\epsilon$ -approchée. L'algorithme  $\mathcal{A}$  est polynomial s'il s'exécute en temps polynomial en  $|I|$ .*

Le facteur, aussi appelé rapport d'approximation ou performance garantie, vérifie  $\epsilon \in ]0, 1[$  pour les problèmes de maximisation et  $\epsilon > 1$  pour les problèmes de minimisation. L'objectif est de trouver un algorithme polynomial qui fournit le rapport d'approximation le plus le proche de 1.

Le rapport d'approximation  $\epsilon$  peut être constant ou dépendre de l'instance  $I$ . Lorsque  $\epsilon$  est constant, l'algorithme  $\mathcal{A}$  est dit à rapport constant. La classe des problèmes d'optimisation admettant un algorithme à rapport constant est notée  $\text{APX}$ .

Lorsque le rapport d'approximation est non-constant, ce dernier peut dépendre d'un paramètre  $\alpha \in \mathbb{X}$  où  $\mathbb{X}$  peut prendre ses valeurs dans  $\mathbb{N}$ ,  $\mathbb{R}$ ,  $[0, 1]$ , etc. Dans ce cas, l'approximation prend la forme d'une fonction  $f : \mathbb{X} \rightarrow ]0, 1[$  pour un problème de maximisation (ou  $f : \mathbb{X} \rightarrow [1, +\infty[$  pour un problème de minimisation). Demko et Hill (1988); Markakis et Psomas (2011) appliquent cette approche pour le problème classique d'allocation de biens indivisibles. Le but est de fournir une

garantie dans le pire cas sur l'utilité des agents selon les paramètres  $\alpha$  et  $\alpha_i$  représentant respectivement l'utilité maximum d'un bien parmi tous les agents et l'utilité maximum de l'agent  $i$  pour un bien (voir la section 2.3, page 62).

Certains problèmes dans  $\text{APX}$  admettent de très bons rapports d'approximation qui se rapprochent de 1. Dans ce cas, le défi est de trouver des algorithmes qui fournissent des rapports d'approximation aussi proches de 1 que l'on souhaite. Ces algorithmes sont appelés schémas et sont définis de la façon suivante.

**Définition 1.3 (Schémas d'approximation)**

- *Un problème d'optimisation  $\pi$  admet un schéma d'approximation polynomial (PTAS pour Polynomial Time Approximation Scheme) si pour tout  $\epsilon \in ]0, 1[$ , il existe un algorithme  $\mathcal{A}_\epsilon$  polynomial en  $|I|$  qui est  $(1-\epsilon)$ -approché pour un problème de maximisation (ou pour tout  $\epsilon > 0$ , l'algorithme est  $(1+\epsilon)$ -approché pour un problème de minimisation).*
- *Un schéma d'approximation complètement polynomial (FPTAS pour Fully Polynomial Time Approximation Scheme) est un PTAS dont la complexité en temps est bornée par un polynôme en  $|I|$  et en  $\frac{1}{\epsilon}$ .*

Les problèmes d'optimisation admettant des schémas d'approximation sont considérés comme les problèmes les moins *difficiles* de la classe NP-difficile. Quant aux problèmes qui n'admettent pas de schémas, en plus de chercher le rapport d'approximation qui soit le plus proche de 1, on peut s'intéresser aux limites de cette approximation, c'est-à-dire montrer l'impossibilité d'obtention de certains rapports d'approximation. Ceci constitue l'étude de l'inapproximabilité d'un problème.

L'inapproximabilité consiste à montrer qu'obtenir un certain rapport d'approximation  $\alpha \in ]0, 1[$  pour un problème de maximisation (ou  $\alpha > 1$  pour un problème de minimisation) contredirait une hypothèse de complexité (par exemple  $\mathbb{P} \neq \text{NP}$ ). Il est donc peu probable qu'il existe un algorithme polynomial  $\alpha$ -approché. Il est alors intéressant de connaître la valeur du plus petit  $\alpha$  d'un problème de maximisation (ou le plus grand  $\alpha$  d'un problème de minimisation) pour lequel le problème est inapproximable. Ceci peut être naturellement fait au moyen de réductions comme la *KARP*-réduction utilisée surtout pour des problèmes de décision. Un autre exemple est la *GAP*-réduction utilisée initialement par Sahni et Gonzalez (1976) pour montrer l'inexistence de  $f(n)$ -approximations pour le TSP (*Traveling Salesman Problem* ou problème du voyageur de commerce, voir le problème 24, page 200) pour toute fonction  $f$  calculable en temps polynomial. Il y a également la *L*-réduction introduite par Papadimitriou et Yannakakis (1991). Pour plus d'informations sur l'inapproximabilité des problèmes d'optimisation, se référer à Hochbaum (1997); Paschos (2004).

Il existe d'autres mesures de l'approximation comme l'erreur relative définie par  $\frac{|OPT(I)-VAL(I,x)|}{OPT(I)}$  que l'on cherche à minimiser, c'est-à-dire que le but est de réduire la distance entre la solution trouvée et l'optimum. Cette mesure est équivalente au rapport d'approximation classique à une bijection près. En effet, en partant du rapport d'approximation classique :  $\frac{VAL(I,x)}{OPT(I)} \geq \epsilon$  pour un problème de maximisation (ou  $\frac{VAL(I,x)}{OPT(I)} \leq \epsilon$  pour un problème de minimisation), on peut déduire l'erreur relative définie par  $\frac{OPT(I)-VAL(I,x)}{OPT(I)} \leq \epsilon'$  pour un problème de maximisation (ou  $\frac{VAL(I,x)-OPT(I)}{OPT(I)} \leq \epsilon'$  pour un problème de minimisation) en posant  $\epsilon' = 1 - \epsilon$  pour tout  $\epsilon \in ]0, 1[$  pour un problème de maximisation (ou  $\epsilon' = \epsilon - 1$  pour tout  $\epsilon > 1$  pour un problème de minimisation).

Une autre mesure, fort intéressante en approximation, est la mesure absolue définie par  $|VAL(I, x) - OPT(I)|$ . Le but est de chercher un algorithme polynomial qui fournit, pour toute instance  $I$ , une solution  $x \in X_I$  telle que  $|OPT(I) - VAL(I, x)| \leq \epsilon$  avec  $\epsilon > 0$ . Très peu de problèmes d'optimisation admettent une telle constante  $\epsilon > 0$ . Par exemple, la coloration des arêtes d'un graphe (problème 12, page 199) utilise au moins  $\Delta$  couleurs où  $\Delta$  est le degré maximum des sommets du graphe. Mais elle peut utiliser au plus  $\Delta + 1$  couleurs (Vizing, 1964) qui est une borne au plus à 1 de l'optimum. D'autres exemples sont mentionnés par Hochbaum (1997).

On cite également le rapport différentiel qui tient compte, non seulement de la solution optimale du problème d'optimisation, mais également de la pire solution. Notons  $PIR(I)$  la valeur de la pire solution d'un problème d'optimisation  $\pi$ , c'est-à-dire la valeur d'une solution optimale lorsque l'on interchange l'objectif du problème ; minimum en maximum et inversement. Le rapport différentiel est défini par  $\frac{|VAL(I,x)-PIR(I)|}{|OPT(I)-PIR(I)|}$  qui s'améliore en se rapprochant de 1, c'est-à-dire qu'on essaie de se rapprocher de l'optimum ou de s'éloigner de la pire solution. Monnot *et al.* (2003) et Paschos (2004) abordent ces mesures de façon détaillée.

### 1.1.2 Conception d'algorithmes d'approximation

Il existe plusieurs techniques pour concevoir des algorithmes d'approximation dont les plus connues sont : les gloutons, la programmation dynamique, la recherche locale, la programmation mathématique, les algorithmes aléatoires, etc. Les lecteurs intéressés peuvent se référer à Hochbaum (1997); Cormen *et al.* (2001); Vazirani (2001); Williamson et Shmoys (2011).

En approximation, les algorithmes gloutons sont souvent les premiers algorithmes que l'on cherche à appliquer. Globalement, les gloutons opèrent de la façon suivante : on part d'une solution non réalisable (par exemple d'une solution vide), puis on prend une série de décisions irrévocables qui modifient la solution jusqu'à

obtenir une solution réalisable. Les gloutons sont souvent simples à construire et à analyser.

Le glouton peut fournir, dans certains cas, une solution exacte comme pour les bases d'un matroïde dont la définition fera l'objet de la section 1.3. D'ailleurs, Edmonds caractérise les matroïdes comme la classe des problèmes dont le glouton fournit une solution optimale (Schrijver, 2003). Dans les prochains chapitres, nous présenterons quelques algorithmes d'approximation combinatoires et gloutons avec garantie de performance pour des problèmes définis sur un matroïde.

Dans d'autres cas, les gloutons peuvent fournir la meilleure approximation qu'un problème puisse avoir comme pour le problème de la *k-couverture de poids maximum* (max *k-coverage*, voir le problème 18, page 199). En partant d'une solution vide, le glouton consiste à choisir, à *k* reprises, le sous-ensemble dont le poids des éléments nouvellement couverts est maximum. Cet algorithme fournit une  $(1 - \frac{1}{e})$ -approximation (Nemhauser *et al.*, 1978) qui est la meilleure approximation à rapport constant que ce problème puisse avoir (Feige, 1998).

Il est également possible d'obtenir des résultats d'approximation via des réductions préservant l'approximation comme la *A-réduction* (Orponen et Mannila, 1987) et la *réduction continue* (Simon, 1990) qui préservent l'appartenance à  $\text{APX}$ . Ou encore la *P-réduction* (Orponen et Mannila, 1987) qui préserve les schémas d'approximation et la *F-réduction* qui préserve l'appartenance à la classe FPTAS. Toutes ces réductions sont reprises en détail par Monnot *et al.* (2003).

Pour plus d'approfondissement sur les algorithmes d'approximation, se référer à Hochbaum (1997); Ausiello (1999); Vazirani (2001); Monnot *et al.* (2003); Paschos (2004); Williamson et Shmoys (2011).

## 1.2 Approximation en optimisation multi-objectif

Soit  $\pi_k$  un problème d'optimisation avec  $k \geq 2$  objectifs. Pour une instance  $I$  de  $\pi_k$ , notons  $X_I$  l'ensemble des solutions réalisables de  $I$ . Pour toute solution  $x \in X_I$ , notons  $VAL_i(I, x)$  la valeur de cette solution sur l'objectif  $i$  pour  $i = 1, \dots, k$ . Nous supposons que  $VAL_i(I, x) \geq 0$  pour tout  $x \in X_I$  et pour toute instance  $I$  de  $\pi_k$ . Dans ce qui suit, notons  $OPT_i(I)$  la valeur optimale de l'objectif  $i$  ( $i = 1, \dots, k$ ).

Les définitions de la section 1.1 sont valables pour des problèmes mono-objectifs, mais elles ne sont pas adaptées pour des problèmes multi-objectifs. En effet, en approximation on évalue la valeur d'une solution approchée par rapport à l'optimum. En optimisation mono-objectif, l'image de l'optimum est unique, ce qui n'est pas le cas en optimisation multi-objectif, puisqu'il n'existe pas toujours de solution réalisable qui optimise tous les objectifs simultanément. Comme alternative, on peut s'intéresser à l'optimalité au sens de Pareto qui a été introduite en



1909 (dans le manuel d'économie politique de Vilfredo Pareto).

Soit  $I$  une instance d'un problème d'optimisation  $\pi_k$  avec  $k \geq 2$  objectifs où l'objectif  $i$  ( $i = 1, \dots, k$ ) est à maximiser (resp. à minimiser).

**Définition 1.4 (Pareto-dominance faible)** *Une solution  $x \in X_I$  est Pareto-dominée faiblement par une solution  $y \in X_I$  si et seulement si  $VAL_i(I, y) \geq VAL_i(I, x)$  (resp.  $VAL_i(I, y) \leq VAL_i(I, x)$ ) pour tout  $i \in \{1, \dots, k\}$ .*

**Définition 1.5 (Pareto-dominance)** *Une solution  $x \in X_I$  est Pareto-dominée par une solution  $y \in X_I$  si et seulement si*

$$\begin{cases} \forall i \in \{1, \dots, k\}, & VAL_i(I, y) \geq VAL_i(I, x) \text{ (resp. } VAL_i(I, y) \leq VAL_i(I, x)), \\ \exists j \in \{1, \dots, k\}, & VAL_j(I, y) > VAL_j(I, x) \text{ (resp. } VAL_j(I, y) < VAL_j(I, x)). \end{cases}$$

**Définition 1.6 (Pareto-optimalité)** *Une solution  $x \in X_I$  est Pareto-optimale s'il n'existe pas d'autre solution  $y \in X_I$  qui Pareto-domine  $x$ .*

En d'autres termes, une solution réalisable est Pareto-optimale s'il n'est pas possible d'améliorer cette solution sur un objectif sans la dégrader sur un autre.

On appelle ensemble de Pareto (ou front de Pareto), l'ensemble des images des solutions Pareto-optimales d'un problème multi-objectif. On ne retient généralement qu'une solution par image optimale. Notons  $PAR_I \subseteq X_I$  l'ensemble des solutions Pareto-optimales de l'instance  $I$ . Pour plus de détails sur l'optimisation multi-objectif en général et sur l'optimalité au sens de Pareto, se référer à Ehrgott (2005); Galand (2008).

Soient  $n$  constantes  $b_1, \dots, b_n$ . Décider s'il existe une solution réalisable  $x \in X_I$  telle que  $VAL_i(I, x) \geq b_i$  (ou  $VAL_i(I, x) \leq b_i$ ),  $i = 1, \dots, k$  est souvent  $\mathbb{NP}$ -complet même si la version mono-objectif du problème est résoluble en temps polynomial. C'est le cas par exemple de la version multi-objectif des problèmes de l'arbre couvrant de poids maximum ou minimum (problème 5, page 198) (Aggarwal *et al.*, 1982; Camerini *et al.*, 1984), du plus court chemin (problème 10, page 198) (Serafini, 1987) et du couplage de poids maximum (problème 13, page 199) (Berger *et al.*, 2011). De plus, certains problèmes peuvent admettre un nombre important, voire exponentiel de solutions Pareto-optimales comme pour le problème de l'arbre de poids maximum/minimum multi-objectif (Hamacher et Ruhe, 1994), ce qui rend impossible leur énumération en temps polynomial.

Ainsi, l'approximation polynomiale est un bon compromis entre la qualité des solutions à retourner et le temps de résolution. Il paraît alors nécessaire d'adapter les définitions classiques de l'approximation aux problèmes multi-objectifs. Mais là encore, différentes approches existent pour l'approximation en multi-objectif. Dans ce qui suit, nous présentons quelques approches, assez répandues dans la

littérature, pour approcher les problèmes multi-objectifs. Pour une description plus détaillée de ces approches, voir par exemple la thèse de Gourvès (2005) ou encore Angel *et al.* (2007). D'autres approches d'approximation en optimisation multi-objectif sont abordées dans Glaßer *et al.* (2010)

### 1.2.1 Approche budget ( $\epsilon$ – constraint)

Il s'agit de transformer tous les objectifs, sauf 1, en contraintes de budget. On obtient un problème mono-objectif à approcher. Soit  $\pi_k$  un problème avec  $k \geq 2$  objectifs où chaque objectif  $i \in \{1, \dots, k\}$  consiste à trouver une solution réalisable  $x \in X_I$ , pour toute instance  $I$ , qui optimise  $VAL_i(I, x)$ . Le problème de budget associé est celui qui optimise  $VAL_i(I, x)$  pour un certain  $i \in \{1, \dots, k\}$ , sous contraintes que pour tout  $j \in \{1, \dots, k\} \setminus \{i\}$ ,  $VAL_j(I, x) \geq B_j$  si l'objectif  $j$  est à maximiser (ou  $VAL_j(I, x) \leq B_j$  si l'objectif  $j$  est à minimiser) pour des budgets  $B_j$ ,  $j \in \{1, \dots, k\} \setminus \{i\}$  donnés. Le problème mono-objectif obtenu est dit à  $(k - 1)$ -budgets.

Les concepts de la section 1.1 peuvent directement être appliqués pour approcher les problèmes d'optimisation à  $p$ -budgets pour  $p \geq 1$ . Nous explorerons cette approche pour un problème bi-objectif défini sur un matroïde dans le chapitre 6 (section 6.5.2, page 161).

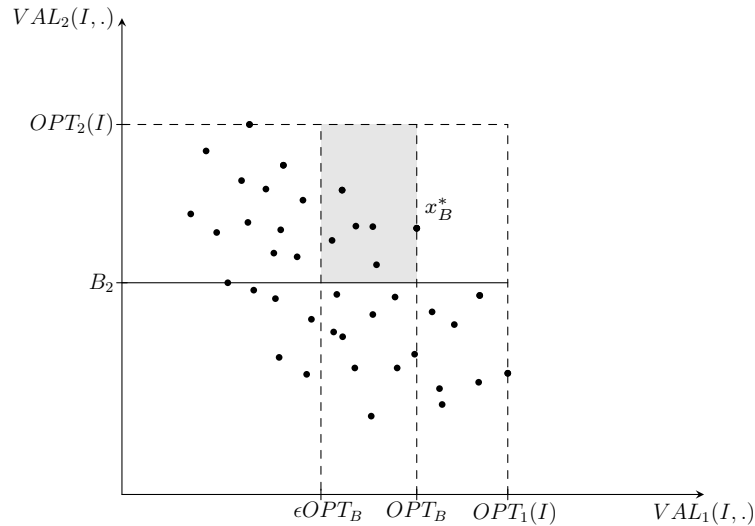


FIGURE 1.2 – Approche budget pour un problème de maximisation bi-objectif.

La figure 1.2 représente l'espace des objectifs d'un problème de maximisation bi-objectif. Les points représentent les images des solutions réalisables. Le second objectif est transformé en contrainte de budget, c'est-à-dire que la valeur

d'une solution réalisable sur le second objectif doit être au moins  $B_2$  (c'est-à-dire,  $VAL_2(I, x) \geq B_2$ ). Le point  $x_B^*$  représente l'optimum du problème de budget de valeur  $OPT_B$ . La zone grisée contient les images des solutions qui  $\epsilon$ -approchent le problème de budget.

Lorsque  $p = 1$  (lorsqu'on associe au problème mono-objectif une seule contrainte de budget), il existe des PTAS pour les problèmes de l'arbre couvrant de poids minimum (Ravi et Goemans, 1996; Hassin et Levin, 2004), du plus court chemin (Warburton, 1987; Hassin, 1992; Lorenz et Raz, 2001) et du couplage maximum (Berger *et al.*, 2011).

À défaut d'avoir des résultats d'approximation pour les problèmes de budget, on peut permettre de violer les contraintes de budget à un petit facteur multiplicateur. Ce type d'approximations est connu dans la littérature en tant qu'*approximation multi-critère* (Grandoni et Zenklusen, 2010).

Par exemple, Hong *et al.* (2004) montrent l'appartenance du problème de l'arbre couvrant avec  $p = 1$  budget à la classe FPTAS multi-critère, lorsqu'on permet de violer la contrainte de budget à un facteur  $1 + \epsilon$  près pour une contrainte dont l'objectif initial est à minimiser, pour tout  $\epsilon > 0$ . Ce résultat a été généralisé à  $p \geq 1$  contraintes de budget par Grandoni *et al.* (2009) qui montrent que lorsque  $p$  est fixé, les problèmes de l'arbre couvrant et du couplage maximum dans un graphe biparti avec  $p$ -budgets appartiennent à la classe PTAS multi-critère.

Pour des discussions plus détaillées sur l'approximation et l'existence de schémas d'approximation en multi-objectif, se référer à Safer et Orlin (1995).

### 1.2.2 Approximation de l'ensemble de Pareto

Cette approche consiste à déterminer un algorithme polynomial qui retourne un ensemble restreint de solutions réalisables qui approchent l'ensemble de Pareto avec une garantie de performance. Cette approche est d'autant plus intéressante lorsqu'il y a un nombre exponentiel de solutions Pareto-optimales. Ici, la notion d'optimalité de Pareto est au cœur du processus d'optimisation.

**Définition 1.7 (Solution approchée en multi-objectif)** *Soit  $I$  une instance d'un problème  $\pi_k$  à  $k$  objectifs. Une solution  $x \in X_I$  est une  $(\epsilon_1, \dots, \epsilon_k)$ -approximation d'une solution  $x^* \in X_I$  si  $\frac{VAL_i(I, x)}{VAL_i(I, x^*)} \geq \epsilon_i$  pour un objectif  $i$  à maximiser avec  $\epsilon_i \in ]0, 1[$  (ou  $\frac{VAL_i(I, x)}{VAL_i(I, x^*)} \leq \epsilon_i$  pour un objectif  $i$  à minimiser avec  $\epsilon_i > 1$ ) pour  $i = 1, \dots, k$ .*

On dit également que la solution  $x$   $(\epsilon_1, \dots, \epsilon_k)$ -domine  $x^*$  (Galand, 2008).

La figure 1.3 représente l'espace des objectifs d'un problème de maximisation bi-objectif. Les points représentent les images des solutions réalisables. Une  $(\epsilon_1, \epsilon_2)$ -

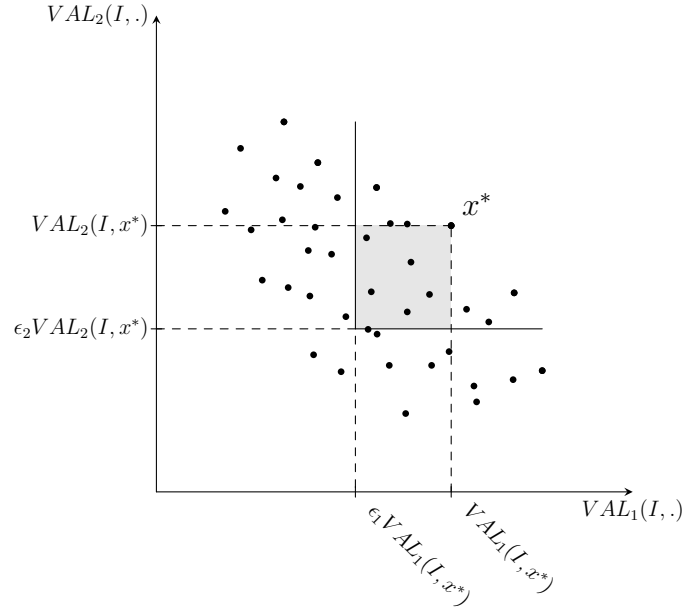


FIGURE 1.3 – Approximation d’une solution en multi-objectif.

approximation du point  $x^*$  est une solution dont l’image se trouve dans la zone grisée.

**Définition 1.8 (Approximation de l’ensemble de Pareto)** Soit  $I$  une instance de  $\pi_k$ . Un ensemble  $P_I \subseteq X_I$  de solutions réalisables est  $(\epsilon_1, \dots, \epsilon_k)$ -approché de l’ensemble de Pareto si pour toute solution réalisable  $y \in X_I$ , il existe  $x \in P_I$  qui  $(\epsilon_1, \dots, \epsilon_k)$ -approche  $y$ .

Lorsque  $\epsilon = \min_{i=1, \dots, k} \epsilon_i$  pour un problème de maximisation multi-objectif (ou  $\epsilon = \max_{i=1, \dots, k} \epsilon_i$  pour un problème de minimisation multi-objectif) alors l’ensemble  $P_I$  est dit  $\epsilon$ -approché de l’ensemble de Pareto de  $I$ .

Remarquons que dans la définition 1.8, nous aurions pu prendre  $y \in PAR(I)$  au lieu de  $y \in X_I$ . En effet, puisque toute solution  $y \in X_I \setminus PAR(I)$  est Pareto-dominée par une solution  $y' \in PAR(I)$ , alors une solution qui  $(\epsilon_1, \dots, \epsilon_k)$ -approche  $y'$  est aussi une  $(\epsilon_1, \dots, \epsilon_k)$ -approximation de  $y$ .

**Définition 1.9 (Algorithme d’approximation de l’ensemble de Pareto)** Un algorithme  $\mathcal{A}$  est  $(\epsilon_1, \dots, \epsilon_k)$ -approché de l’ensemble de Pareto si pour toute instance  $I$  de  $\pi_k$ , l’algorithme  $\mathcal{A}$  produit, en temps polynomial en  $|I|$ , un ensemble de solutions réalisables  $P_I \subseteq X_I$  qui est  $(\epsilon_1, \dots, \epsilon_k)$ -approché.

L’approximation de l’ensemble de Pareto avec un ensemble de solutions est un compromis intéressant. Cela permet de réduire la difficulté du problème en

déterminant un ensemble de solutions de taille raisonnable. De plus, plusieurs solutions sont fournies, ce qui permet au décideur de visualiser les solutions de rechange.

Cette approche a suscité un grand intérêt lorsque Papadimitriou et Yannakakis (2000) ont montré dans leur article fondateur que pour tout problème d'optimisation multi-objectif  $\pi_k$  avec un nombre d'objectif fixé et pour tout instance  $I$  de  $\pi_k$ , il existe toujours un ensemble de solutions  $P_I$  polynomial en  $|I|$  et en  $\frac{1}{\epsilon}$  qui est  $\epsilon$ -approché de l'ensemble de Pareto pour tout  $\epsilon > 0$  (Papadimitriou et Yannakakis, 2000). Cependant, il n'existe pas toujours d'algorithme polynomial qui permette d'obtenir cet ensemble de solutions. Une condition nécessaire et suffisante pour l'existence d'un tel algorithme est l'existence d'un algorithme polynomial qui résout le problème GAP (voir le problème 14, page 199). En utilisant cette condition, Papadimitriou et Yannakakis (2000) montrent l'existence de FPTAS pour construire des  $\epsilon$ -approximations de l'ensemble de Pareto pour les versions multi-objectifs des problèmes du plus court chemin, de l'arbre couvrant de poids minimum et du couplage.

Une autre conséquence des résultats de Papadimitriou et Yannakakis (2000) est la possibilité d'obtenir une approximation de l'ensemble de Pareto à partir d'une approximation du problème de budget et vice versa. En effet, à partir d'une approximation du problème de budget, on peut construire un ensemble de solutions qui approchent l'ensemble de Pareto en subdivisant l'intervalle des valeurs des objectifs mis en budget sur une échelle logarithmique. Puis pour chaque subdivision, sélectionner la solution qui approche le problème de budget correspondant. Inversement, à partir d'une approximation de l'ensemble de Pareto, et pour un vecteur budget donné, on peut résoudre le problème GAP associé (Papadimitriou et Yannakakis, 2000, Théorème 2) qui fournit une approximation au problème de budget mais il est possible de violer certaines contraintes de budget. En effet, il suffit de choisir la solution approchée qui approche le mieux l'objectif du problème de budget et qui ne viole pas les contraintes de budget, si possible.

D'un point de vue décisionnel, l'éventualité de fournir des ensembles  $P_I$  avec des tailles arbitrairement grandes (même si la taille de  $P_I$  reste toujours polynomiale en  $I$  et en  $\frac{1}{\epsilon}$ ) peut rendre difficile le choix de solutions au décideur. Pour pallier ce problème, Vassilvitskii et Yannakakis (2005) posent le problème d'approcher l'ensemble de Pareto avec des ensembles  $P_I$  aussi petits que possible. Ils proposent des algorithmes d'approximation généraux avec des bornes sur la taille de  $P_I$  pour toute instance  $I$  pour des problèmes à 2 et à 3 objectifs. La thèse de Jamain (2014) traite également de la cardinalité de l'ensemble des solutions approchées, ainsi que de sa *stabilité*, c'est-à-dire qu'il ne contient pas de solutions approchées redondantes.

En l'absence de processus décisionnel, une alternative serait de proposer une unique solution de compromis. On peut alors s'intéresser à approcher le problème multi-objectif avec une seule solution, en approchant par exemple le point idéal du problème qui est le principe de la prochaine section.

### 1.2.3 Approximation du point idéal

En optimisation multi-objectif, le point idéal est une solution, pas nécessairement réalisable, mais qui atteint l'optimum sur tous les objectifs simultanément. Cela signifie que l'image du point idéal vaut  $(OPT_1(I), OPT_2(I), \dots, OPT_k(I))$ . Bien que cette solution n'existe pas toujours, on peut tout de même rechercher une solution réalisable qui approche, au mieux, ce point idéal. La définition suivante propose une approximation du point idéal, appelée aussi approche simultanée dans Angel *et al.* (2007).

**Définition 1.10 (Solution approchée du point idéal)** Soit  $I$  une instance d'un problème multi-objectif  $\pi_k$ . Une solution  $x \in X_I$  est  $(\epsilon_1, \dots, \epsilon_k)$ -approchée du point idéal si  $\frac{VAL_i(I,x)}{OPT_i(I)} \geq \epsilon_i$  pour un objectif  $i$  à maximiser avec  $\epsilon_i \in ]0, 1[$  (ou  $\frac{VAL_i(I,x)}{OPT_i(I)} \leq \epsilon_i$  pour un objectif  $i$  à minimiser avec  $\epsilon_i > 1$ ) pour tout  $i \in \{1, \dots, k\}$ .

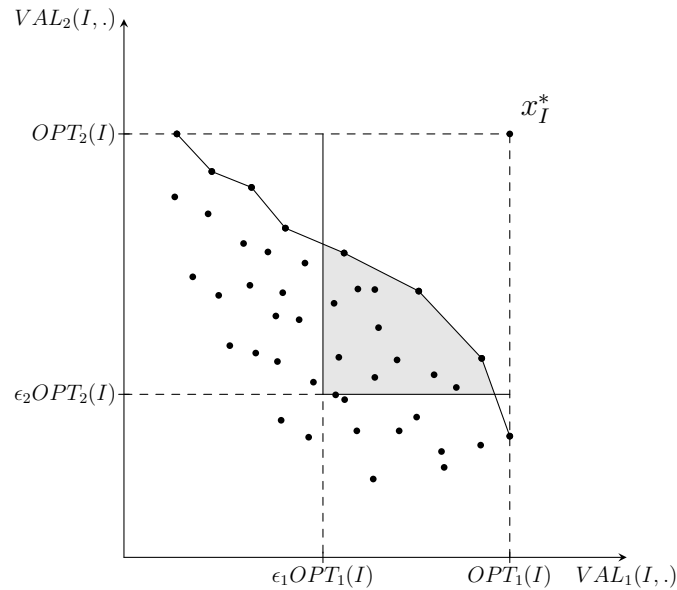


FIGURE 1.4 – Approximation du point idéal pour un problème de maximisation bi-objectif.

La figure 1.4 représente l'espace des objectifs d'un problème de maximisation bi-objectif. Les points représentent les images des solutions réalisables et le front

de Pareto est caractérisé par l'ensemble des points reliés par des segments. Le point  $x_I^*$  représente le point idéal du problème. La zone grisée contient tous les points qui  $(\epsilon_1, \epsilon_2)$ -approchent le point idéal  $x_I^*$ .

Lorsque  $\epsilon = \min_{i=1, \dots, k} \epsilon_i$  pour un problème de maximisation multi-objectif (ou  $\epsilon = \max_{i=1, \dots, k} \epsilon_i$  pour un problème de minimisation multi-objectif) alors la solution est dite  $\epsilon$ -approchée du point idéal.

**Définition 1.11 (Algorithme d'approximation du point idéal)**

Un algorithme  $\mathcal{A}$  est  $(\epsilon_1, \dots, \epsilon_k)$ -approché du point idéal si pour toute instance  $I$  de  $\pi_k$ , l'algorithme  $\mathcal{A}$  retourne, en temps polynomial en  $|I|$ , une solution réalisable  $x_I \in X_I$  qui  $(\epsilon_1, \dots, \epsilon_k)$ -approche le point idéal.

Cette approche est appliquée entre autres par Hansen (1980); Ravi *et al.* (1993); Stein et Wein (1997); Angel *et al.* (2005); Goel et Meyerson (2006); Bazgan *et al.* (2013); Escoffier *et al.* (2013). Par exemple, Escoffier *et al.* (2013) proposent un algorithme qui retourne une solution  $\frac{1}{3}$ -approchée (ou  $(\frac{1}{3}, \frac{1}{3})$ -approchée) du point idéal du problème de l'arbre de poids maximum bi-objectif. Pour  $k \geq 3$  objectifs, il existe des instances pour lesquelles il n'existe aucune  $\alpha$ -approximation du point idéal quel que soit  $\alpha > 0$  (Escoffier *et al.*, 2013).

On remarque que l'approximation du point idéal ne fait pas intervenir la notion de Pareto-optimalité car chaque objectif est considéré de manière indépendante. Pourtant, nous observons avec la remarque suivante qu'approcher l'ensemble de Pareto avec une unique solution est équivalent à approcher le point idéal avec cette même solution.

**Remarque 1.1** Soit  $I$  une instance d'un problème  $\pi_k$  avec  $k \geq 2$  objectifs à maximiser. Soit  $x \in X_I$  une solution réalisable de  $I$ . Alors  $\{x\}$  est  $(\epsilon_1, \dots, \epsilon_k)$ -approché de l'ensemble de Pareto de  $I$  si et seulement si  $x$  est  $(\epsilon_1, \dots, \epsilon_k)$ -approchée du point idéal de  $I$ .

**Preuve.** Supposons que  $\{x\}$  est  $(\epsilon_1, \dots, \epsilon_k)$ -approché de l'ensemble de Pareto de  $I$ . Alors d'après la définition 1.8 (page 19), pour tout  $y \in X_I$ ,  $\frac{VAL_i(I, x)}{VAL_i(I, y)} \geq \epsilon_i$  pour  $i = 1, \dots, k$ . Cela reste vrai en particulier pour  $y_i \in X_I$  tel que  $VAL_i(I, y_i) = OPT_i(I)$ . D'où,  $\frac{VAL_i(I, x)}{OPT_i(I)} \geq \epsilon_i$  pour  $i = 1, \dots, k$ . Donc  $x$  est  $(\epsilon_1, \dots, \epsilon_k)$ -approchée du point idéal de  $I$  d'après la définition 1.10 (page 21).

Pour la réciproque, supposons que  $x$  est  $(\epsilon_1, \dots, \epsilon_k)$ -approchée du point idéal de  $I$ . Alors d'après la définition 1.10,  $\frac{VAL_i(I, x)}{OPT_i(I)} \geq \epsilon_i$  pour  $i = 1, \dots, k$ . Pour un problème de maximisation, toute solution  $y \in X_I$  satisfait  $VAL_i(I, y) \leq OPT_i(I)$ , alors pour tout  $y \in X_I$ ,  $\frac{VAL_i(I, x)}{VAL_i(I, y)} \geq \epsilon_i$  pour  $i = 1, \dots, k$  donc  $\{x\}$  est  $(\epsilon_1, \dots, \epsilon_k)$ -approché de l'ensemble de Pareto de  $I$  d'après la définition 1.8 (page 19). ■

Notons que la remarque 1.1 est également valide pour des objectifs à minimiser.

Parmi toutes les solutions qui approchent le point idéal, celle qui offre le meilleur compromis entre les objectifs, c'est-à-dire la solution qui fournit l'approximation  $\epsilon = \min_{i=1,\dots,k} \epsilon_i$  pour un problème de maximisation multi-objectif (ou  $\epsilon = \max_{i=1,\dots,k} \epsilon_i$  pour un problème de minimisation multi-objectif) la plus proche de 1, est surnommée le *point d'articulation* (*knee*) de l'ensemble de Pareto par Vassilvitskii et Yannakakis (2005). L'obtention de cette solution est NP-difficile mais ils proposent un FPTAS pour approcher le point d'articulation sous condition qu'il existe un algorithme polynomial pour résoudre le problème GAP associé (problème 14, page 199). Ceci est vérifié pour les versions multi-objectifs du plus court chemin, de l'arbre couvrant et du couplage (Vassilvitskii et Yannakakis, 2005).

#### 1.2.4 Inapproximabilité en optimisation multi-objectif

En multi-objectif, la notion d'inapproximabilité peut prendre plusieurs sens. L'impossibilité de fournir une approximation peut être due à la structure du problème où certains résultats d'approximation peuvent s'avérer en contradiction avec des hypothèses de complexité bien établies (par exemple  $\mathbb{P} \neq \mathbb{NP}$ ). L'inapproximabilité peut alors être décelée au moyen de réductions comme en optimisation mono-objectif.

Mais il est également possible que certaines zones de l'espace des objectifs ne contiennent pas du tout de solutions réalisables. Avec l'approximation de la courbe de Pareto, il existe toujours un ensemble de solutions  $P_I$  pour toute instance  $I$  qui soit  $\epsilon$ -approché pour tout  $\epsilon > 0$  (Papadimitriou et Yannakakis, 2000). Mais si on impose une taille  $p \geq 1$  à cet ensemble de solutions, alors il n'existe pas toujours d'ensemble  $P_I$  de taille  $|P_I| \leq p$  pour toute instance  $I$  qui soit  $\epsilon$ -approché. Il suffit donc de trouver une instance  $I$  pour laquelle il n'existe pas d'ensemble  $P_I$  qui contienne  $|P_I| \leq p$  solutions réalisables et qui fournisse un certain rapport d'approximation de l'ensemble de Pareto (ou du point idéal si  $p = 1$ ).

L'objectif de cette thèse est de fournir une solution de compromis pour des problèmes multi-objectifs (ou multi-agents). Le principe de compromis adopté dans les chapitres 4, 5 et 6 de ce manuscrit est l'approximation du point idéal du problème. Cette approche permet de fournir une borne inférieure sur l'utilité de chaque agent pour la solution de compromis. Cette borne inférieure est aussi appelée *garantie dans le pire cas*. Nous fournirons également quelques résultats d'inapproximabilité en donnant des bornes supérieures sur le meilleur rapport d'approximation qu'on puisse avoir. Ces techniques seront appliquées à une classe particulière de problèmes d'optimisation combinatoire multi-objectifs, il s'agit des problèmes qui ont une structure de matroïde.



## 1.3 Théorie des matroïdes

La théorie des matroïdes est apparue dès le début du XX<sup>e</sup> siècle sous diverses formulations axiomatiques (Schrijver, 2003). Le terme matroïde a été introduit par le mathématicien Hassler Whitney en 1935 dans l'article pionnier Whitney (1935). L'étude des matroïdes a été motivée par le concept d'indépendance en algèbre linéaire. Par la suite, la théorie des matroïdes a contribué de manière significative à la compréhension de certaines structures combinatoires importantes telles que les forêts d'un multi-graphe, le couplage dans un graphe biparti, etc. Nous présentons dans cette section quelques notions sur les matroïdes. Pour plus de détails, les lecteurs intéressés peuvent consulter Oxley (1992); Gondran et Minoux (1995); Schrijver (2003); Korte et Vygen (2008).

Nous utilisons souvent certaines abréviations sur les notations des opérations ensemblistes : soit l'ensemble  $F$ , alors  $F + e$  et  $F - e$  font référence respectivement à  $F \cup \{e\}$  et  $F \setminus \{e\}$ .

### 1.3.1 Généralités sur les matroïdes

**Définition 1.12 (Matroïde)** *Korte et Vygen (2008)* Un matroïde  $\mathcal{M} = (X, \mathcal{F})$  est un système d'ensembles défini par un ensemble d'éléments  $X$  et une collection  $\mathcal{F}$  de sous-ensembles de  $X$  vérifiant les axiomes suivants :

- (i) **non vide** :  $\emptyset \in \mathcal{F}$  ;
- (ii) **hérédité** : si  $F_j \subseteq F_i$  et  $F_i \in \mathcal{F}$  alors  $F_j \in \mathcal{F}$  ;
- (iii) **augmentation** : pour tout couple  $F_1, F_2 \in \mathcal{F}$  tel que  $|F_1| < |F_2|$ , il existe  $x \in F_2 \setminus F_1$  tel que  $F_1 \cup \{x\} \in \mathcal{F}$ .

Par induction, (iii) est équivalent à

- (iii') pour tout couple  $F_1, F_2 \in \mathcal{F}$  tel que  $|F_1| < |F_2|$ , il existe  $A \subseteq F_2 \setminus F_1$  avec  $|A| = |F_2| - |F_1|$  tel que  $F_1 \cup A \in \mathcal{F}$ .

Les éléments de  $\mathcal{F}$  sont dits *ensembles indépendants* et tous les autres éléments de  $2^X \setminus \mathcal{F}$  sont dits *ensembles dépendants*.

**Exemple 1.1** Soit  $X = \{e_1, e_2, e_3, e_4\}$ . Considérons les collections suivantes :

- $\mathcal{F}_1 = \{\emptyset, \{e_1\}, \{e_3\}, \{e_1, e_2\}\}$ ,
- $\mathcal{F}_2 = \{\emptyset, \{e_1\}, \{e_2\}, \{e_3\}, \{e_1, e_2\}\}$ ,
- $\mathcal{F}_3 = \{\emptyset, \{e_1\}, \{e_2\}, \{e_3\}, \{e_1, e_2\}, \{e_1, e_3\}\}$ ,
- $\mathcal{F}_4 = \{\emptyset, \{e_1\}, \{e_2\}, \{e_3\}, \{e_4\}, \{e_1, e_2\}, \{e_1, e_3\}, \{e_2, e_3\}, \{e_2, e_4\}, \{e_3, e_4\}, \{e_1, e_2, e_3\}, \{e_2, e_3, e_4\}\}$ .

La structure  $(X, \mathcal{F}_1)$  n'est pas un matroïde puisque l'axiome (ii) de la définition 1.12 n'est pas vérifié, du fait que  $\{e_1, e_2\} \in \mathcal{F}_1$  mais que  $\{e_2\} \notin \mathcal{F}_1$ . De même,  $(X, \mathcal{F}_2)$  n'est pas un matroïde car l'axiome (iii) n'est pas vérifié. En effet, puisque

$\{e_3\} \in \mathcal{F}_2$  et  $\{e_1, e_2\} \in \mathcal{F}_2$  alors d'après l'axiome (iii), il existe  $e \in \{e_1, e_2\}$  tel que  $\{e, e_3\} \in \mathcal{F}_2$ , or un tel ensemble n'existe pas dans  $\mathcal{F}_2$ . Enfin, il est facile de vérifier que  $(X, \mathcal{F}_3)$  et  $(X, \mathcal{F}_4)$  définissent bien des matroïdes.

On appelle *oracle d'indépendance* l'oracle qui détermine si un ensemble donné  $F \subseteq X$  est indépendant ou non pour un matroïde  $\mathcal{M} = (X, \mathcal{F})$ . La complexité de cet oracle, notée  $\theta$ , dépend du matroïde considéré. Nous supposons dans cette thèse que  $\theta$  est toujours polynomial.

Les sous-ensembles dépendants minimaux au sens de l'inclusion sont appelés *circuits*. Pour  $X' \subseteq X$ , les sous-ensembles indépendants maximaux au sens de l'inclusion qui sont dans  $X'$  sont appelés *bases* de  $X'$ . Lorsque  $X' = X$ , ces sous-ensembles sont des bases du matroïde  $\mathcal{M}$ .

**Définition 1.13 (Base d'un matroïde)** *Une base d'un matroïde  $\mathcal{M} = (X, \mathcal{F})$  est un ensemble indépendant  $B \in \mathcal{F}$  tel qu'il n'existe pas  $B' \in \mathcal{F}$  avec  $B \subset B'$ .*

La collection des bases de  $X$  est notée  $\mathcal{B}$ .

**Remarque 1.2** *Toutes les bases  $B \in \mathcal{B}$  sont de même cardinalité.*

**Preuve.** Supposons qu'il existe deux bases  $B_1$  et  $B_2$  telles que  $|B_1| < |B_2|$ . En vertu de l'axiome (iii) des matroïdes, il existe un élément  $e \in B_2 \setminus B_1$  tel que  $B_1 \cup \{e\} \in \mathcal{F}$ , contradiction avec  $B_1$  est une base (ensemble indépendant maximal pour l'inclusion). ■

**Exemple 1.2** Dans l'exemple 1.1, les bases du matroïde  $(X, \mathcal{F}_3)$  sont les ensembles  $\{e_1, e_2\}$  et  $\{e_1, e_3\}$ . La collection des bases du matroïde  $(X, \mathcal{F}_4)$  est  $\mathcal{B} = \{\{e_1, e_2, e_3\}, \{e_2, e_3, e_4\}\}$ .

Un matroïde peut également être caractérisé par la collection de ses bases :

**Théorème 1.1 (Axiome des bases) Whitney (1935)** *Soit  $X$  un ensemble et soit  $\mathcal{B}$  une collection non vide de sous-ensembles de  $X$ . Alors les résultats suivants sont équivalents :*

- (a)  $\mathcal{B}$  est la collection des bases d'un matroïde ;
- (b) si  $B_1, B_2 \in \mathcal{B}$  et  $e_2 \in B_2 \setminus B_1$ , alors  $B_2 - e_2 + e_1 \in \mathcal{B}$  pour un certain  $e_1 \in B_1 \setminus B_2$  ;
- (c) si  $B_1, B_2 \in \mathcal{B}$  et  $e_2 \in B_2 \setminus B_1$ , alors  $B_1 - e_1 + e_2 \in \mathcal{B}$  pour un certain  $e_1 \in B_1 \setminus B_2$ .

La cardinalité des bases d'un matroïde définit le *rang*  $r(\mathcal{M})$  du matroïde  $\mathcal{M}$ .

La fonction rang d'un matroïde  $\mathcal{M} = (X, \mathcal{F})$  est la fonction  $r : 2^X \rightarrow \mathbb{N}$  définie par  $r(U) = \max\{|F| : F \in \mathcal{F}, F \subseteq U\}$  pour  $U \subseteq X$ . Un matroïde est aussi caractérisé par sa fonction rang :

**Théorème 1.2 (Axiome du rang)** *Whitney (1935)* Soit  $X$  un ensemble d'éléments et soit la fonction  $r : 2^X \rightarrow \mathbb{N}$ . La fonction  $r$  est le rang d'un matroïde  $\mathcal{M} = (X, \mathcal{F})$  où  $\mathcal{F} = \{F \subseteq X : r(F) = |F|\}$  si et seulement si elle vérifie les axiomes suivants :

1. **vide** :  $r(\emptyset) = 0$  ;
2. **hérédité** : si  $F_1 \subseteq F_2$  alors  $r(F_1) \leq r(F_2)$  ;
3. **sous-modularité** :  $r(F_1) + r(F_2) \geq r(F_1 \cup F_2) + r(F_1 \cap F_2)$ .

### Matroïdes simples

La définition suivante caractérise une certaine sous-classe de matroïdes appelée les *matroïdes simples* :

**Définition 1.14 (Matroïde simple)** *Oxley (1992)* Un matroïde est dit simple s'il ne contient pas de circuit comprenant 1 ou 2 éléments.

La définition 1.14 signifie qu'un matroïde est simple si tout élément ou tout couple d'éléments est indépendant.

**Exemple 1.3** Dans l'exemple 1.1 (page 24), le matroïde  $(X, \mathcal{F}_3)$  n'est pas simple car les ensembles  $\{e_4\}$ ,  $\{e_1, e_4\}$ ,  $\{e_2, e_3\}$ ,  $\{e_2, e_4\}$  et  $\{e_3, e_4\}$  ne sont pas indépendants. La même remarque est faite pour le matroïde  $(X, \mathcal{F}_4)$  car l'ensemble  $\{e_1, e_4\}$  est dépendant.

Nous proposons dans ce qui suit une généralisation de la notion de matroïde simple.

**Définition 1.15 (Matroïde  $k$ -simple)** Un matroïde est dit  $k$ -simple pour un certain entier  $k \geq 1$  si tout sous-ensemble d'au plus  $k$  éléments est indépendant.

La notion initiale de matroïde simple correspond au cas où  $k = 2$  (2-simple).

### Restriction, contraction et troncature

Soient un matroïde  $\mathcal{M} = (X, \mathcal{F})$  et un sous-ensemble  $X' \subseteq X$ .

- La *restriction* de  $\mathcal{M}$  à  $X'$ , notée  $\mathcal{M}|_{X'}$ , est la structure  $(X', \mathcal{F}')$  où  $\mathcal{F}' = \{F \in \mathcal{F} : F \subseteq X'\}$ .
- Si  $X' \in \mathcal{F}$  alors la *contraction*<sup>1</sup> de  $\mathcal{M}$  par  $X'$ , notée  $\mathcal{M}/X'$ , est la structure  $(X \setminus X', \mathcal{F}'')$  où  $\mathcal{F}'' = \{F \subseteq X \setminus X' : F \cup X' \in \mathcal{F}\}$ .
- Soit  $k \in \mathbb{N}$ . La  *$k$ -troncature* de  $\mathcal{M}$ , notée  $\mathcal{M}_{\leq k}$ , est la structure  $(X, \mathcal{F}_{\leq k})$  où  $\mathcal{F}_{\leq k} = \{F \in \mathcal{F} : |F| \leq k\}$ .

---

1. La contraction est définie dans la littérature pour tout  $X' \subseteq X$ , et lorsque  $X' \in \mathcal{F}$  la définition est similaire à celle présentée dans ce manuscrit.

Il est bien connu que  $\mathcal{M}|_{X'}$ ,  $\mathcal{M}/X'$  et  $\mathcal{M}_{\leq k}$  sont des matroïdes (Schrijver, 2003).

**Exemple 1.4** Soit le matroïde  $\mathcal{M} = (X, \mathcal{F}_4)$  défini dans l'exemple 1.1 (page 24).

- La restriction de  $\mathcal{M}$  à  $X' = \{e_1, e_2\}$  est le matroïde  $\mathcal{M}|_{X'} = (X', \mathcal{F}'_4)$  où  $\mathcal{F}'_4 = \{\emptyset, \{e_1\}, \{e_2\}, \{e_1, e_2\}\}$ .
- La contraction de  $\mathcal{M}$  à  $X'' = \{e_4\}$  est le matroïde  $\mathcal{M}/X'' = (X \setminus X'', \mathcal{F}''_4)$  où  $\mathcal{F}''_4 = \{\emptyset, \{e_2\}, \{e_3\}, \{e_2, e_3\}\}$ .
- La 2-troncature de  $\mathcal{M}$  est le matroïde  $\mathcal{M}_{\leq 2} = (X, \mathcal{F}'''_4)$  où  $\mathcal{F}'''_4 = \{\emptyset, \{e_1\}, \{e_2\}, \{e_3\}, \{e_4\}, \{e_1, e_2\}, \{e_1, e_3\}, \{e_2, e_3\}, \{e_2, e_4\}, \{e_3, e_4\}\}$ .

### Bases interchangeables

Les bases d'un matroïde satisfont les propriétés d'échange suivantes.

**Théorème 1.3 (Propriété d'échange d'un élément)** *Brunaldi (1969)*

*Soient  $A$  et  $B$  deux bases d'un matroïde  $\mathcal{M}$  et soit  $a \in A \setminus B$ . Alors il existe  $b \in B \setminus A$  tel que  $A - a + b$  et  $B + a - b$  sont des bases de  $\mathcal{M}$ .*

**Théorème 1.4 (Propriété d'échange d'un ensemble)** *Brylawski (1973);*

*Greene (1973); Woodall (1974)* *Soient  $A$  et  $B$  des bases d'un matroïde  $\mathcal{M}$  et soit  $\{A_1, A_2\}$  une bi-partition de  $A$ . Alors il existe une bi-partition  $\{B_1, B_2\}$  de  $B$  telle que  $A_1 \cup B_2$  et  $A_2 \cup B_1$  sont des bases de  $\mathcal{M}$ .*

Le théorème 1.4 est existentiel mais la construction de  $\{B_1, B_2\}$  peut se faire en temps polynomial en utilisant l'algorithme d'intersection de deux matroïdes qui sera présenté dans la section 1.3.3 (Bixby et Cunningham, 1995).

**Théorème 1.5 (Propriété d'échange multiple d'ensembles)**

*Greene et Magnanti (1975)* *Soient  $A$  et  $B$  des bases d'un matroïde  $\mathcal{M}$ , et soit  $\{A_1, \dots, A_n\}$  une partition de  $A$ . Alors il existe une partition  $\{B_1, \dots, B_n\}$  de  $B$  telle que  $A \setminus A_i \cup B_i$ ,  $1 \leq i \leq n$  sont toutes des bases de  $\mathcal{M}$ .*

## 1.3.2 Exemples de matroïdes

Il existe plusieurs exemples de matroïdes qui permettent de modéliser des problèmes connus en optimisation combinatoire. Nous présentons dans ce qui suit quelques exemples de matroïdes ainsi que des applications potentielles.

### Matroïde matriciel (ou linéaire)

Une collection  $\mathcal{F}$  contenant les sous-ensembles de colonnes linéairement indépendantes d'une matrice réelle forme le *matroïde matriciel*. Il s'agit

du point de départ de la théorie des matroïdes, qui a par la suite généralisé la notion d'indépendance linéaire à d'autres structures comme les graphes.

### Matroïde graphique

Soit  $G = (V, E)$  un multi-graphe non orienté. Une forêt  $F \subseteq E$  est un sous-ensemble d'arêtes de  $G$  sans cycles. Notons  $\mathcal{F}$  l'ensemble des forêts de  $G$ . La structure  $(E, \mathcal{F})$  définit un matroïde appelé le *matroïde graphique*. Vérifions les axiomes qui définissent un matroïde selon la définition 1.12 page 24 :

1.  $\emptyset$  est une forêt ;
2. si  $F$  est une forêt, alors tout sous-ensemble  $F' \subseteq F$  est aussi une forêt ;
3. soient deux forêts  $F_1$  et  $F_2$  telles que  $|F_1| < |F_2|$ . Soit une arête  $e$  de  $F_2$  qui diminue le nombre de composantes connexes de  $F_1$ . Alors  $F_1 \cup \{e\}$  est aussi une forêt.

Une base du matroïde graphique est une forêt maximale pour l'inclusion, ce qui correspond à un arbre couvrant si le multi-graphe  $G$  est connexe.

Si  $G$  est simple (s'il ne contient pas de boucles et qu'il existe au plus une arête entre tout couple de sommets distincts), alors le matroïde graphique est simple. En effet, dans un graphe simple, une ou deux arêtes ne peuvent constituer de cycle.

Le matroïde graphique correspond également au matroïde matriciel lorsque la matrice utilisée représente la matrice d'incidence d'un graphe (Gondran et Minoux, 1995).

Le matroïde graphique peut avoir des applications dans la conception de réseaux de communication. Les sommets sont assimilés à des nœuds de connexion et les arêtes représentent les liens possibles entre les nœuds. Une connexion minimale entre les nœuds est un arbre couvrant du graphe qui représente une base du matroïde graphique si le graphe est connexe.

### Matroïde libre

La collection  $\mathcal{F}$  de tous les sous-ensembles d'un ensemble  $X$  forme le *matroïde libre*. Cela est équivalent à dire que  $\mathcal{F} = 2^X$ .

### Matroïde $k$ -uniforme

La collection de tous les sous-ensembles de taille au plus  $k$  d'un ensemble  $X$  forme le *matroïde  $k$ -uniforme*.

Le matroïde  $k$ -uniforme peut modéliser des problèmes d'élection à vainqueurs multiples en choix social. En effet, les éléments représentent les candidats. On peut définir des votants qui vont exprimer leurs préférences sur les candidats. Un ensemble indépendant de ce matroïde peut représenter l'ensemble des vainqueurs si on tolère qu'il y ait au plus  $k$  vainqueurs. Ces vainqueurs peuvent être déterminés en appliquant une règle de vote (Procaccia *et al.*, 2008; Lu et Boutilier, 2011; Skowron *et al.*, 2013).

### Matroïde de couplage

Étant donné un graphe non orienté  $G = (V, E)$ , soit  $\mathcal{F}$  la collection des sous-ensembles de  $V$  qui peuvent être couverts par un couplage<sup>2</sup> de  $G$ , alors  $(V, \mathcal{F})$  est le *matroïde de couplage* associé à  $G$  (Edmonds et Fulkerson, 1965).

### Matroïde transversal

Soient  $m$  sous-ensembles  $X_1, \dots, X_k$ , non nécessairement disjoints, d'un ensemble  $X$ . Un *transversal partiel* est un ensemble  $T \subseteq X$  tel qu'il existe une fonction injective  $\Phi : T \rightarrow [k]$  de sorte que  $t \in X_{\Phi(t)}$  pour tout  $t \in T$ . Cela signifie que  $T$  contient au plus un élément par ensemble  $X_i$  ( $i = 1, \dots, k$ ). La collection  $\mathcal{F} = \{T \subseteq X : T \text{ est un transversal partiel de } X\}$  constitue le matroïde transversal. Une base, appelée aussi transversal, est un ensemble qui contient exactement un élément par sous-ensemble  $X_i$  ( $i = 1, \dots, k$ ).

Il est possible de représenter ce problème dans un graphe biparti. L'ensemble des sommets de gauche correspond aux éléments  $e_i$ ,  $i = 1, \dots, m$  de  $X$ , et les sommets de droite correspondent aux sous-ensembles  $X_j$ ,  $j = 1, \dots, k$ . Il existe une arête  $(e_i, X_j)$  si et seulement si  $e_i \in X_j$ . Un sous-ensemble  $T \subseteq X$  est un transversal partiel s'il existe un couplage qui couvre les sommets de  $T$  (Gondran et Minoux, 1995).

**Exemple 1.5** Considérons le problème d'ordonnancement qui consiste à exécuter un ensemble  $X$  de tâches de durée unitaire sur une machine. Chaque tâche  $e \in X$  a une date butoir  $d(e)$  représentant son heure de début d'exécution au plus tard. L'objectif est d'associer une heure de début d'exécution  $t(e)$  à chaque tâche  $e \in X$  de sorte que  $t(e) \leq d(e)$

---

2. Un couplage dans un graphe est un sous ensemble d'arêtes disjointes.

pour tout  $e \in X$  et  $t(e) \neq t(e')$  pour toutes les tâches  $e \neq e'$ . Ce problème peut être modélisé par le matroïde transversal où les éléments  $X$  sont les tâches et tel que le sous-ensemble  $X_i$  contient toutes les tâches  $e \in X$  dont la date butoir  $d(e) \leq i$  pour  $i = 1, \dots, \max_{e \in X} d(e)$ .

D'autres applications du matroïde transversal sur des problèmes d'ordonnement sont mentionnées dans Maffioli *et al.* (2007).

### Matroïde des chemins disjoints

Soit un graphe orienté  $G = (V, E)$  et soit  $v \in V$  un sommet de  $G$ . La collection  $\mathcal{F}$  de tous les sous-ensembles de sommets  $V' \subseteq V$  tels qu'il existe des chemins arête-disjoints entre  $v$  et chaque sommet de  $V'$  forme le *matroïde des chemins disjoints*.

Ce matroïde peut avoir des applications dans les problèmes de sécurité. Par exemple, pour la conception d'issues de secours disjointes dans de grandes infrastructures pour l'évacuation de la population en cas d'incident.

### Matroïde de partition

Soit  $X$  un ensemble d'éléments partitionné en  $k$  ensembles  $X_1, \dots, X_k$  pour un certain entier  $k \geq 1$ . Cela signifie que  $\bigcup_{i=1}^k X_i = X$  et  $X_i \cap X_j = \emptyset$  pour tout  $i \neq j$ . Soient  $k$  entiers positifs  $b_1, \dots, b_k$ . Les ensembles  $F \subseteq X$  satisfaisant  $|F \cap X_i| \leq b_i$  forment le matroïde de partition.

#### Exemple 1.6 (Allocation de biens indivisibles)

Allouer un ensemble  $S = \{e_1, \dots, e_m\}$  de biens indivisibles à  $n$  agents peut être vu comme un *matroïde de partition*. Il suffit de poser  $X_i = \{e_i^1, e_i^2, \dots, e_i^n\}$  et  $b_i = 1$  pour tout  $i \in [m]$ . Le fait de choisir un élément  $e_i^k$  signifie qu'on alloue l'objet  $e_i$  à l'agent  $k$ . Ce problème est repris en détail dans la section 2.1.1 (page 48).

### Matroïde laminaire

Une collection  $\mathcal{L}$  de sous-ensembles de  $X$  est une *famille laminaire* (ou *hiérarchique*) si pour tout  $A, B \in \mathcal{L}$ ,  $A \subseteq B$  ou  $B \subseteq A$  ou  $A \cap B = \emptyset$ . Soit  $b(A)$  un entier positif associé à chaque  $A \in \mathcal{L}$ . Soit  $\mathcal{F} = \{F \subseteq X : |F \cap A| \leq b(A), \forall A \in \mathcal{L}\}$ , alors  $\mathcal{M} = (X, \mathcal{F})$  est le matroïde laminaire associé à  $\mathcal{L}$ .

Le *matroïde laminaire* généralise le matroïde de partition puisqu'une partition  $\{X_1, \dots, X_k\}$  de  $X$  vérifie  $X_i \cap X_j = \emptyset$  pour tout  $i \neq j$ . Ainsi,  $\{X_1, \dots, X_k\}$  est bien une famille laminaire.

Une application possible du matroïde laminaire est la formation d'un groupe représentant une large communauté qui est structurée en une hiérarchie et de sorte que le nombre de membres de chaque unité de la hiérarchie soit fixé.

**Exemple 1.7** Le syndicat d'une entreprise est constitué d'un groupe de travailleurs de différents services. Par exemple, les services de la compagnie sont  $\mathcal{L} = \{\text{service financier (SF)}, \text{comptabilité (C)}, \text{ressources humaines (RH)}, \text{recherche et développement (RD)}, \text{services généraux (SG)}, \text{logistique (L)}, \text{gestion des risques (GR)}\}$  où  $C \subset \text{SF}$ ,  $L \subset \text{SG}$ ,  $\text{GR} \subset \text{SG}$ , et l'intersection de tous les autres couples de services est vide.  $\mathcal{L}$  définit bien une famille laminaire. Soit  $X$  l'ensemble des travailleurs de la compagnie et  $b(S)$  un entier positif représentant le nombre de membres du service  $U \in \mathcal{L}$  dans le syndicat. Un ensemble indépendant du matroïde laminaire est un syndicat de la compagnie qui contient au plus  $b(U)$  membres par service  $U \in \mathcal{L}$ .

### 1.3.3 Matroïdes pondérés

Soit une fonction  $u : X \rightarrow \mathbb{R}^+$ . Un *matroïde pondéré*  $\mathcal{M} = (X, \mathcal{F}, u)$  est un matroïde où chaque élément  $e \in X$  a un poids (ou utilité)  $u(e) \geq 0$ . Par abus de notation, on notera  $u(X')$  le poids (ou l'utilité) du sous-ensemble  $X' \subseteq X$ . Si  $u(X') = \sum_{x \in X'} u(x)$  pour tout  $X' \subseteq X$  alors la fonction  $u$  est dite *additive*. Dans ce qui suit, nous supposons que  $u$  est additive.

#### Problème de l'ensemble indépendant de poids maximum (ou base de poids minimum)

Étant donné un matroïde pondéré  $\mathcal{M} = (X, \mathcal{F}, u)$ , un problème d'optimisation classique consiste à trouver un ensemble indépendant  $F \in \mathcal{F}$  de poids  $u(F)$  maximum (ou une base  $B \in \mathcal{B}$  de poids  $u(B)$  minimum).

Dans le cas d'un problème de maximisation et puisque la fonction  $u$  est positive, une solution optimale est toujours une base  $B \in \mathcal{B}$ . On appellera souvent ce problème « le problème de la base de poids maximum ». Si l'objectif est de minimiser le poids d'un ensemble indépendant, alors l'ensemble vide sera toujours la solution optimale. Ainsi, il conviendrait de définir les bases comme seules solutions réalisables.

Il existe un algorithme GLOUTON qui résout le problème de la base de poids maximum (ou minimum) en temps polynomial (Edmonds, 1971) et qui est décrit dans l'algorithme 1.1. Le GLOUTON généralise l'algorithme de Kruskal pour le problème de l'arbre couvrant de poids maximum (ou minimum).



---

**Algorithme 1.1 : GROUTON (Edmonds, 1971)**

---

**Données** : un matroïde pondéré  $\mathcal{M} = (X, \mathcal{F}, u)$ ,  $obj \in \{\min, \max\}$   
**Résultat** : une base de poids minimum / maximum

- 1 **si**  $obj = \min$  **alors**
- 2 |   renommer  $X = \{e_1, \dots, e_{|X|}\}$  de sorte que  $u(e_i) \leq u(e_{i+1})$ ,  $i = 1, \dots, |X| - 1$
- 3 **sinon**
- 4 |   renommer  $X = \{e_1, \dots, e_{|X|}\}$  de sorte que  $u(e_i) \geq u(e_{i+1})$ ,  $i = 1, \dots, |X| - 1$
- 5  $B \leftarrow \emptyset$
- 6 **pour**  $i = 1$  à  $|X|$  **faire**
- 7 |   **si**  $B \cup \{e_i\} \in \mathcal{F}$  **alors**
- 8 |   |    $B \leftarrow B \cup \{e_i\}$
- 9 **retourner**  $B$

---

La complexité du GROUTON (algorithme 1.1) est en  $O(|X| \max\{\ln |X|, \theta\})$  où  $\theta$  est la complexité de l'oracle d'indépendance que nous supposons polynomiale et qui intervient à l'étape 7 de l'algorithme 1.1.

Par souci de simplicité, on suppose que la collection  $\mathcal{F}$  est en entrée dans nos algorithmes. Mais cette collection peut être représentée par l'oracle d'indépendance. Ainsi, même si  $\mathcal{F}$  est de taille exponentielle, l'oracle d'indépendance permet de vérifier l'indépendance d'un sous-ensemble d'éléments en temps polynomial (selon notre hypothèse).

Soient un matroïde  $\mathcal{M} = (X, \mathcal{F})$ , un ensemble indépendant  $F \in \mathcal{F}$  et une fonction additive  $u : X \rightarrow \mathbb{R}^+$ .

- Le *complément* de  $F$ , noté  $\mathbf{C}(\mathcal{M}, F)$ , consiste en tous les ensembles  $E \subseteq X$  tels que  $F \cup E$  est une base de  $\mathcal{M}$ .
- Le *complément maximum* (resp. *minimum*) de  $F$  pour  $u$ , noté  $\mathbf{C}^{\max}(\mathcal{M}, F, u)$  (resp.  $\mathbf{C}^{\min}(\mathcal{M}, F, u)$ ), consiste en tous les ensembles  $E \subseteq X$  tels que  $F \cup E$  est une base de  $\mathcal{M}$  et  $u(E)$  est maximum (resp. minimum).

Nous supposons que  $\mathbf{C}(\mathcal{M}, B)$  n'est jamais vide. En particulier,  $\mathbf{C}(\mathcal{M}, B) = \{\emptyset\}$  pour toute base  $B$  de  $\mathcal{M}$ . Un élément de  $\mathbf{C}^{obj}(\mathcal{M}, F, u)$  avec  $obj \in \{\min, \max\}$  peut être obtenu en appliquant GROUTON  $(\mathcal{M}/F, u, obj)$  où  $\mathcal{M}/F$  est la contraction de  $\mathcal{M}$  par  $F$ .

Dans ce qui suit, nous nous intéressons uniquement à des problèmes de maximisation. Le poids d'une base maximum pour une fonction  $u$  est noté  $OPT_u(\mathcal{M})$ . La collection des bases maximums pour  $u$  est notée  $\mathcal{B}_u$ .

Les lemmes suivants sont des résultats originaux et seront utilisés dans les prochains chapitres.

**Lemme 1.1** *Soit  $S$  un ensemble indépendant d'un matroïde pondéré  $\mathcal{M} = (X, \mathcal{F}, u)$  où  $OPT_u(\mathcal{M}) \geq \rho_0$  et  $OPT_u(\mathcal{M}/S) < \rho_1 \leq \rho_0$ . Alors pour toute base  $T$  de  $\mathcal{M}/S$ ,*

$$OPT_u(\mathcal{M}/T) \geq \rho_0 - \rho_1.$$

**Preuve.** Soit  $T$  une base de  $\mathcal{M}/S$  et soit  $F \in \mathcal{B}_u$ , c'est-à-dire  $u(F) = OPT_u(\mathcal{M})$ . Puisque  $S \cup T$  est une base de  $\mathcal{M}$ , alors d'après le théorème 1.4 (page 27),  $F$  peut être partitionné en  $\{F_1, F_2\}$  tel que  $F_1 \cup T$  et  $S \cup F_2$  sont des bases de  $\mathcal{M}$ . Il s'ensuit que  $OPT_u(\mathcal{M}/S) \geq u(F_2)$  et  $OPT_u(\mathcal{M}/T) \geq u(F_1)$ . Par conséquent,  $OPT_u(\mathcal{M}/S) + OPT_u(\mathcal{M}/T) \geq u(F_1) + u(F_2) = OPT_u(\mathcal{M})$ . Finalement,  $\rho_0 \leq OPT_u(\mathcal{M}) \leq OPT_u(\mathcal{M}/S) + OPT_u(\mathcal{M}/T) < \rho_1 + OPT_u(\mathcal{M}/T)$ , donc  $OPT_u(\mathcal{M}/T) \geq \rho_0 - \rho_1$ . ■

**Lemme 1.2** Soit  $\mathcal{M} = (X, \mathcal{F}, u)$  un matroïde pondéré et soient  $F_1, F_2 \in \mathcal{F}$  tels que  $|F_1| < |F_2|$ . Supposons que  $F_2 = \{e_1, \dots, e_{|F_2|}\}$  avec  $u(e_1) \geq \dots \geq u(e_{|F_2|})$ . Dans le matroïde contracté  $\mathcal{M}/F_1$ , il existe  $E \subseteq F_2 \setminus F_1$  tel que  $E$  est indépendant dans  $\mathcal{M}/F_1$  (c'est-à-dire  $F_1 \cup E \in \mathcal{F}$ ) où  $|E| = |F_2| - |F_1|$  et  $u(E) \geq u(\{e_{|F_1|+1}, \dots, e_{|F_2|}\})$ .

**Preuve.** Par récurrence sur  $k = |F_2| - |F_1|$ , montrons qu'il existe un ensemble indépendant satisfaisant ce lemme.

Pour  $k = 1$ , en utilisant l'axiome (iii) de la définition 1.12 des matroïdes (page 24), il existe  $e \in F_2 \setminus F_1$  tel que  $F_1 \cup \{e\} \in \mathcal{F}$ . Puisque  $u(e_1) \geq \dots \geq u(e_{|F_2|})$ , alors  $u(e) \geq u(e_{|F_2|})$ . D'où,  $E_1 = \{e\}$ .

Nous supposons que le lemme 1.2 est vrai pour  $k \geq 1$  et nous montrons sa validité pour  $k + 1$ . Soient  $F_1, F_2 \in \mathcal{F}$  tels que  $|F_2| - |F_1| = k + 1$ ,  $F_2 = \{e_1, \dots, e_{|F_1|+k+1}\}$  et  $u(e_1) \geq \dots \geq u(e_{|F_1|+k+1})$ . Soit  $F'_2 = F_2 \setminus \{e_{|F_1|+k+1}\}$ . En utilisant l'hypothèse de récurrence, il existe  $E_k \subseteq F'_2 \setminus F_1$  dans le matroïde contracté  $\mathcal{M}/F_1$  tel que  $F_1 \cup E_k \in \mathcal{F}$ ,  $|E_k| = |F'_2| - |F_1| = k$  et  $u(E_k) \geq u(\{e_{|F_1|+1}, \dots, e_{|F_1|+k}\})$ . Considérons les ensembles  $F_1 \cup E_k$  et  $F_2$  où  $|F_1 \cup E_k| = |F_1| + k$  et  $|F_2| = |F_1| + k + 1$ . De l'axiome (iii) des matroïdes, il existe  $e \in F_2 \setminus (F_1 \cup E_k)$  tel que  $F_1 \cup E_k \cup \{e\} \in \mathcal{F}$ . En d'autres termes,  $E_k \cup \{e\}$  est un ensemble indépendant de  $\mathcal{M}/F_1$ . Puisque  $u(e_1) \geq \dots \geq u(e_{|F_1|+k+1})$  alors,  $u(e) \geq u(e_{|F_1|+k+1})$ . Soit  $E_{k+1} = E_k \cup \{e\}$ . Par conséquent,  $u(E_{k+1}) \geq u(\{e_{|F_1|+1}, \dots, e_{|F_2|}\})$ . ■

Le lemme 1.2 est existentiel mais il est possible de construire l'ensemble  $E$  du lemme 1.2 en temps polynomial. Initialement,  $E = \emptyset$ , puis trier les éléments de  $F_2 \setminus F_1$  par ordre décroissant selon  $u$ . En suivant cet ordre, ajouter un élément de  $F_2 \setminus (F_1 \cup E)$  à  $E$  si  $F_1 \cup E$  est indépendant. Ceci est fait en  $O(|X| \max\{\ln |X|, \theta\})$ .

### Problèmes d'intersection de deux matroïdes

**Définition 1.16** Korte et Vygen (2008) Étant donnés deux matroïdes  $(X, \mathcal{F}_1)$  et  $(X, \mathcal{F}_2)$ , leur intersection est définie par  $(X, \mathcal{F}_1 \cap \mathcal{F}_2)$ .

Notons que l'intersection de deux matroïdes n'est pas forcément un matroïde.

Il existe des algorithmes plus élaborés que le GROUTON pour résoudre les problèmes suivants en temps polynomial :

### I. Intersection de deux matroïdes de cardinalité maximum

Soient deux matroïdes  $\mathcal{M}_1 = (X, \mathcal{F}_1)$  et  $\mathcal{M}_2 = (X, \mathcal{F}_2)$ . Le but est de trouver un ensemble indépendant commun  $F \in \mathcal{F}_1 \cap \mathcal{F}_2$  tel que  $|F|$  est maximum.

#### Exemple 1.8 (Couplage dans un graphe biparti) (Gondran et Minoux, 1995)

Soit un graphe biparti  $G = (L \cup R, E)$  où  $L = \{\ell_1, \dots, \ell_{|L|}\}$  et  $R = \{r_1, \dots, r_{|R|}\}$ . Soit  $\delta_G(s) \subseteq E$  le cocycle du sommet  $s$ , c'est-à-dire l'ensemble des arêtes incidentes à  $s \in L \cup R$  dans  $G$ . Un transversal<sup>3</sup> à gauche (resp. à droite) est un sous-ensemble d'arêtes  $F \subseteq E$  tel que pour tout sommet  $s \in L$  (resp.  $s \in R$ ),  $|\delta_{G'}(s)| \leq 1$  où  $G' = (L \cup R, F)$  est le graphe partiel induit par l'ensemble d'arêtes  $F$ .

Les transversaux à gauche et à droite peuvent être modélisés par les matroïdes de partition respectifs  $\mathcal{M}_L = (E, \mathcal{F}_L)$  et  $\mathcal{M}_R = (E, \mathcal{F}_R)$ . Pour  $\mathcal{M}_L$ , l'ensemble des arêtes  $E$  est partitionné en  $\delta_G(\ell_1), \dots, \delta_G(\ell_{|L|})$  et  $\mathcal{F}_L = \{F \subseteq E : |F \cap \delta_G(\ell_i)| \leq 1, i = 1, \dots, |L|\}$ . Dans le matroïde  $\mathcal{M}_R$ , l'ensemble des arêtes  $E$  est partitionné en  $\delta_G(r_1), \dots, \delta_G(r_{|R|})$  et  $\mathcal{F}_R = \{F \subseteq E : |F \cap \delta_G(r_i)| \leq 1, i = 1, \dots, |R|\}$ .

L'intersection de  $\mathcal{M}_L$  et  $\mathcal{M}_R$  fournit des ensembles  $F \subseteq E$  où  $|\delta_{G'}(s)| \leq 1$  pour tout  $s \in L \cup R$  dans le graphe partiel  $G' = (L \cup R, F)$  induit par  $F$ . Cela correspond à la définition d'un couplage de  $G$ .

#### Exemple 1.9 (Arbre couvrant avec un nombre maximum de couleurs)

(Broersma et Li, 1997) Le problème de l'arbre couvrant avec un nombre maximum de couleurs est défini sur un graphe non orienté et connexe  $G = (V, E)$  tel que chaque arête  $e \in E$  a une couleur  $\mathcal{L}(e) \in \{\ell_1, \dots, \ell_p\}$  pour un entier  $p \geq 1$ . Les couleurs d'un sous-ensemble d'arêtes  $E' \subseteq E$  sont définies par  $\mathcal{L}(E') = \bigcup_{e \in E'} \mathcal{L}(e)$ . L'objectif est de trouver un arbre couvrant  $T$  de  $G$  qui contient un nombre maximum de couleurs  $|\mathcal{L}(T)|$ .

Ce problème revient à résoudre un problème d'intersection de deux matroïdes de cardinalité maximum. Broersma et Li (1997) définissent les deux matroïdes suivants :

- $\mathcal{M}_1$  est le matroïde graphique de  $G$ ,
- $\mathcal{M}_2 = (E, \mathcal{F}_2)$  est le matroïde de partition défini par  $\mathcal{F}_2 = \{E' \subseteq E : |\mathcal{L}(E')| = |E'|\}$ . En effet, les arêtes  $E$  de  $G$  peuvent être partitionnées en  $E_1, \dots, E_p$  de sorte que l'ensemble  $E_i$  ( $i = 1, \dots, p$ ) contienne les arêtes de couleur  $\ell_i$ . Il suffit alors de poser  $b_i = 1$  pour  $i = 1, \dots, p$  dans la définition du matroïde de partition.

---

3. La définition des transversaux à gauche et à droite dans un graphe biparti diffère de la définition du matroïde transversal.

L'intersection de cardinalité maximum de ces deux matroïdes donne la plus grande forêt qui contient des arêtes de couleurs distinctes. Si cette forêt n'est pas un arbre couvrant, alors il suffit de compléter la solution partielle par n'importe quelles arêtes jusqu'à obtenir une solution réalisable (un arbre couvrant).

Reprenons le modèle général défini sur deux matroïdes  $\mathcal{M}_1 = (X, \mathcal{F}_1)$  et  $\mathcal{M}_2 = (X, \mathcal{F}_2)$ , et qui consiste à trouver un ensemble  $F \in \mathcal{F}_1 \cap \mathcal{F}_2$  de cardinalité maximum. L'algorithme d'Edmonds (1979) résout ce problème en  $O(|X|^3\theta)$  et il opère de la façon suivante. En partant de l'ensemble vide  $F = \emptyset$ , la taille de la solution  $F$  est augmentée de 1 à chaque itération de l'algorithme. Pour cela, l'idée consiste à rechercher des chemins alternés dans un graphe auxiliaire qui est construit à chaque itération. Le graphe auxiliaire est biparti et orienté. L'ensemble des sommets de gauche est défini par  $F$  et celui de droite par  $X \setminus F$ . Pour les arcs, soit  $C_i(F, y) = \{x \in F + y : F + y \notin \mathcal{F}_i, F + y - x \in \mathcal{F}_i\}$  l'unique circuit de  $F \cup \{y\}$  qui contient  $x$  dans  $\mathcal{M}_i$  ( $i = 1, 2$ ). Pour tout  $y \in X \setminus F$ , ajouter l'arc  $(x, y)$  si  $x \in C_1(F, y) \setminus \{y\}$  et ajouter l'arc  $(y, x)$  si  $x \in C_2(F, y) \setminus \{y\}$ . Puis, poser  $S_F^i = \{y \in X \setminus F : F \cup \{y\} \in \mathcal{F}_i\}$  pour  $i = 1, 2$ . L'objectif est de rechercher un plus court chemin de  $S_F^1$  à  $S_F^2$  qui permettra d'augmenter la taille de  $F$ . La solution partielle  $F$  est augmentée en lui ajoutant les éléments de droite du chemin et en supprimant ceux de gauche. Si  $S_F^1 \cap S_F^2 \neq \emptyset$ , alors le chemin est de longueur 0, il suffit alors d'augmenter la solution en prenant un élément de  $S_F^1 \cap S_F^2$ . L'algorithme s'arrête lorsqu'il n'y a plus de chemin entre  $S_F^1$  et  $S_F^2$ . Pour plus de détails sur cet algorithme, voir par exemple Korte et Vygen (2008).

## II. Intersection de deux matroïdes de poids maximum

Le problème d'intersection de deux matroïdes pondérés de poids maximum  $\mathcal{M}_1 = (X, \mathcal{F}_1, u)$  et  $\mathcal{M}_2 = (X, \mathcal{F}_2, u)$  consiste à trouver un ensemble indépendant commun  $F \in \mathcal{F}_1 \cap \mathcal{F}_2$  tel que  $u(F)$  est maximum.

Dans l'exemple 1.8 (page 34), si les arêtes sont pondérées alors le problème de couplage de poids maximum dans un graphe biparti revient à résoudre l'intersection des deux matroïdes de partition  $\mathcal{M}_L$  et  $\mathcal{M}_R$  de poids maximum.

**Exemple 1.10 (Arborescence de poids maximum)** (Gondran et Minoux, 1995) Soit  $G = (V, E)$  un graphe orienté et connexe et soit  $r \in V$  une racine. Chaque arc  $e \in E$  est muni d'un poids  $w(e)$ . Un sous-ensemble d'arcs  $\mathcal{T} \subseteq E$  constitue une *arborescence* de racine  $r$  si  $\mathcal{T}$  est un arbre couvrant et chaque sommet  $v \in V$  a au plus un arc incident intérieurement. Une définition équivalente est l'existence, pour tout  $v \in V \setminus \{r\}$ , d'un unique chemin de  $v$  à  $r$  dans le graphe partiel  $(V, \mathcal{T})$ .

L'objectif est de trouver une arborescence  $\mathcal{T} \subseteq E$  de racine  $r$  et de poids  $w(\mathcal{T}) = \sum_{e \in \mathcal{T}} w(e)$  maximum. On peut modéliser ce problème comme l'intersection des deux matroïdes suivants :

- $\mathcal{M}_1 = (E, \mathcal{F}_1)$  est le matroïde graphique associé à  $G$  ;
- $\mathcal{M}_2 = (E, \mathcal{F}_2)$  est un matroïde de partition défini comme suit :  $E$  est partitionné en  $\{\delta^-(1), \dots, \delta^-(|V|)\}$  où  $\delta^-(v)$  est l'ensemble des arcs incidents intérieurement au sommet  $v \in V$  et  $\mathcal{F}_2 = \{F \subseteq E : |F \cap \delta^-(r)| = 0, |F \cap \delta^-(v)| \leq 1, \forall v \in V \setminus \{r\}\}$ .

La recherche d'une arborescence de poids maximum est équivalente à l'intersection des deux matroïdes  $\mathcal{M}_1$  et  $\mathcal{M}_2$  de poids maximum.

Le problème d'intersection de deux matroïdes de poids maximum peut être résolu en temps polynomial avec l'algorithme primal-dual de Frank (1981) en  $O(|X^4| + |X|^3\theta)$ , qui peut être réduit à  $O(|X|^3\theta)$  (Korte et Vygen, 2008). En partant de l'ensemble vide  $F = F_0 = \emptyset$ , l'algorithme de Frank (1981) consiste à augmenter la cardinalité de la solution à chaque itération en construisant des ensembles  $F_0, \dots, F_s \in \mathcal{F}_1 \cap \mathcal{F}_2$  où  $|F_k| = k$  pour  $k = 0, \dots, s$  et  $s = \max\{|F| : F \in \mathcal{F}_1 \cap \mathcal{F}_2\}$ . Les ensembles  $F_k$  sont optimaux, c'est-à-dire  $u(F_k) = \max\{u(F) : F \in \mathcal{F}_1 \cap \mathcal{F}_2, |F| = k\}$ . Il suffit alors de choisir un ensemble parmi  $F_0, \dots, F_s$  de poids maximum. Pour cela, la fonction  $u$  est divisée en deux parties  $u_1$  et  $u_2$  telles que  $u(e) = u_1(e) + u_2(e)$  pour tout  $e \in X$ . Ces fonctions doivent également vérifier  $u_i(F_k) = \max\{u_i(F) : F \in \mathcal{F}_i, |F| = k\}$  pour  $i = 1, 2$ . Initialement,  $u_1(e) = u(e)$  et  $u_2(e) = 0$  pour tout  $e \in X$ . Puis, à chaque itération, un sous-graphe est construit comme dans l'algorithme d'Edmonds (1979) mais en retenant uniquement les arcs dont les extrémités sont de poids maximum parmi celles de  $S_{F_k}^i$ . De même, un chemin avec un nombre minimum d'arcs est recherché. Puis, les poids  $u_1$  et  $u_2$  sont mis à jour en posant  $u_1(x) := u_1(x) - \epsilon$  et  $u_2(x) := u_2(x) + \epsilon$  pour les éléments  $e$  que l'on peut atteindre depuis  $S_{F_k}^1$  où  $\epsilon$  est obtenu en évaluant les poids  $u_1$  et  $u_2$  des éléments restants. Pour plus de détails sur l'algorithme de Frank (1981), voir par exemple Korte et Vygen (2008).

Par contre, de manière générale, le problème d'intersection de  $k \geq 3$  matroïdes de cardinalité maximum est NP-difficile. Cela peut être démontré par une réduction du problème du chemin hamiltonien dans un graphe orienté (problème 11, page 198) ou par une réduction du problème *3-dimensional matching* (problème 1, page 197) (Camerini et Maffioli, 1975).

### Partition de matroïdes pondérés

La propriété d'échange d'un élément (voir le théorème 1.3, page 27) peut être étendue à la propriété *d'échange forte d'un élément* définie sur l'ensemble  $\mathcal{B}_u =$

$\{B \in \mathcal{B} : u(B) = \sum_{e \in B} u(e) \text{ est maximum}\}$  des bases maximums pour  $u$ .

**Propriété 1.1 (Propriété d'échange forte d'un élément)** Soient  $B_1, B_2 \in \mathcal{B}_u$  avec  $B_1 \neq B_2$ . Alors pour tout  $e_1 \in B_1 \setminus B_2$ , il existe  $e_2 \in B_2 \setminus B_1$  tels que  $B_1 - e_1 + e_2$  et  $B_2 - e_2 + e_1$  sont dans  $\mathcal{B}_u$ .

**Preuve.** Puisque  $\mathcal{B}_u \subseteq \mathcal{B}$ , nous savons grâce au théorème 1.3 (page 27) que pour tout  $e_1 \in B_1 \setminus B_2$ , il existe  $e_2 \in B_2 \setminus B_1$  tels que  $B_1 - e_1 + e_2$  et  $B_2 - e_2 + e_1$  sont dans  $\mathcal{B}$ .  $B_1$  appartient à  $\mathcal{B}_u$ , donc  $u(B_1 - e_1 + e_2) \leq u(B_1)$  qui est équivalent à  $u(e_2) \leq u(e_1)$ . De même,  $u(B_2 - e_2 + e_1) \leq u(B_2)$  implique que  $u(e_2) \geq u(e_1)$ . D'où,  $u(e_2) = u(e_1)$ . Par conséquent,  $u(B_1 - e_1 + e_2) = u(B_1)$  et  $u(B_2 - e_2 + e_1) = u(B_2)$ . Ceci signifie que  $B_1 - e_1 + e_2$  et  $B_2 - e_2 + e_1$  sont dans  $\mathcal{B}_u$ . ■

Le résultat suivant montre l'existence d'une partition d'une base  $A^*$  maximum pour  $u$  en  $n \geq 2$  parts à partir de n'importe quelle autre base  $A$  déjà partitionnée en  $n$  parts, tout en garantissant une borne inférieure sur l'utilité des parts de  $A^*$ .

**Théorème 1.6** Soit  $\mathcal{M} = (X, \mathcal{F}, u)$  un matroïde pondéré. Étant donnée une base de poids maximum  $A^* \in \mathcal{B}_u$  et une partition  $\{A_1, \dots, A_n\}$  d'une autre base  $A$ , alors il existe une partition  $\{A_1^*, \dots, A_n^*\}$  de  $A^*$  vérifiant

$$\min_{i \in [n]} u(A_i^*) \geq \min_{i \in [n]} u(A_i)$$

**Preuve.** Par récurrence sur  $|A^* \setminus A|$ . Si  $|A^* \setminus A| = 0$  alors  $A^* = A$  et le théorème est vrai en posant  $A_i^* = A_i$  pour tout  $i \in N$ .

Supposons que  $|A^* \setminus A| = k$  pour un certain  $k > 0$  et que le théorème est vrai pour  $|A^* \setminus A| \in \{0, \dots, k-1\}$ . Puisque  $k \geq 1$ , alors il existe un élément  $x$  dans  $A^* \setminus A$ . En utilisant le théorème 1.3 (page 27), il existe  $s \in A \setminus A^*$  tel que  $A - s + x$  et  $A^* - x + s$  sont des bases. Puisque  $A = A_1 \cup \dots \cup A_n$ , alors  $s \in A_i$  pour un certain  $i \in \{1, \dots, n\}$ . Supposons sans perte de généralité que  $s \in A_1$ . Alors  $|A^* \setminus (A - s + x)| = k - 1$ , donc le théorème est vrai par hypothèse pour les bases  $A^*$ ,  $A' = A - s + x$  et la partition  $\{A_1 - s + x, A_2, \dots, A_n\}$ . Ainsi,  $A^*$  peut être partitionnée en  $A_1^* \cup \dots \cup A_n^*$  de sorte que

$$\min_{i \in [n]} u(A_i^*) \geq \min\{u(A_1 - s + x), u(A_2), \dots, u(A_n)\} \quad (1.1)$$

Si  $u(s) > u(x)$  alors  $u(A^* - x + s) > u(A^*)$ , contradiction avec  $A^* \in \mathcal{B}_u$ . Il s'ensuit que  $u(s) \leq u(x)$ . Donc  $u(A_1 - s + x) \geq u(A_1)$  et

$$\min\{u(A_1 - s + x), u(A_2), \dots, u(A_n)\} \geq \min\{u(A_1), \dots, u(A_n)\}. \quad (1.2)$$

Le théorème pour  $|A^* \setminus A| = k$  découle de (1.1) et (1.2). ■

Soit  $A$  une base partitionnée en  $n$  parts  $\{A_1, \dots, A_n\}$ . Nous montrons dans ce qui suit qu'il existe une permutation  $\sigma$  des  $n$  parts qui permet d'obtenir une garantie (borne inférieure) sur la valeur maximale d'une base, même en contractant les  $i$  premières parts de  $A$ , pour tout  $i \in \{1, \dots, n\}$ .

**Théorème 1.7** *Soit  $A$  une base d'un matroïde pondéré  $\mathcal{M} = (X, \mathcal{F}, u)$  partitionnée en  $\{A_1, \dots, A_n\}$  où  $n \geq 1$ . Il existe une permutation  $\sigma$  de  $\{1, \dots, n\}$  telle que pour tout  $i \in [n]$ ,  $OPT_u(\mathcal{M}/(\cup_{j \leq i} A_{\sigma(j)})) \geq \frac{n-i}{n} OPT_u(\mathcal{M})$ . Par ailleurs, la permutation  $\sigma$  peut se construire en  $O(n^2 |X| \max\{\ln |X|, \theta\})$ .*

**Preuve.** Par récurrence sur  $n$ . Pour  $n = 1$ ,  $A = A_1$  et  $OPT_u(\mathcal{M}/A_1) = 0$ .

Pour  $n \geq 2$ , commençons par  $i = 1$ . On veut montrer que  $OPT_u(\mathcal{M}/A_{\sigma(1)}) \geq \frac{n-1}{n} OPT_u(\mathcal{M})$ . Soit  $B$  une base de  $\mathcal{M}$  d'utilité  $u$  maximum, c'est-à-dire  $u(B) = OPT_u(\mathcal{M})$ . Nous savons grâce au théorème 1.5 (page 27) que  $B$  peut être partitionnée en  $\{B_1, \dots, B_n\}$  où  $A \setminus A_i \cup B_i$  est une base de  $\mathcal{M}$  pour  $i = 1, \dots, n$ . Sans perte de généralité, supposons que  $B_1 = \arg \max_{i \in [n]} u(B_i)$ , donc

$$u(B_1) \geq \frac{1}{n} u(B) = \frac{1}{n} OPT_u(\mathcal{M}) \quad (1.3)$$

Avant de compléter la preuve du théorème 1.7, nous donnons un résultat intermédiaire.

**Lemme 1.3** *Soient  $A, B$  des bases d'un matroïde pondéré  $\mathcal{M} = (X, \mathcal{F}, u)$ . Étant donné un entier  $n \geq 2$ , soient  $\{A_1, \dots, A_n\}$  et  $\{B_1, \dots, B_n\}$  des  $n$ -partitions respectivement de  $A$  et  $B$  telles que  $A \setminus A_i \cup B_i$  est une base de  $\mathcal{M}$  pour  $i = 1, \dots, n$ . Alors pour tout  $i \in \{1, \dots, n\}$ , il existe  $j \in \{1, \dots, n\}$  tel que*

$$OPT_u(\mathcal{M}/A_j) \geq \frac{1}{n-1} ((n-2)u(B) + u(B_i))$$

**Preuve.** Par récurrence sur  $n \geq 2$ . Pour  $n = 2$ , en utilisant le théorème 1.4 (page 27), on sait que  $A_{3-i} \cup B_i$  est une base de  $\mathcal{M}$  pour  $i = 1, 2$ . Par conséquent,  $B_i$  est une base de  $\mathcal{M}/A_{3-i}$ . Il s'ensuit que  $OPT_u(\mathcal{M}/A_{3-i}) \geq u(B_i)$  pour  $i = 1, 2$ .

Pour  $n > 2$ , soit  $i \in \{1, \dots, n\}$ . Dans le matroïde contracté  $\mathcal{M}/B_i$ ,  $A \setminus A_i$  et  $B \setminus B_i$  sont deux bases.  $A \setminus A_i$  est partitionnée en  $\{A_k, k \in \{1, \dots, n\} \setminus \{i\}\}$ . Soit  $\{B'_k, k \in \{1, \dots, n\} \setminus \{i\}\}$  une partition de  $B' = B \setminus B_i$  telle que  $(A \setminus A_i) \setminus A_k \cup B'_k$  est une base de  $\mathcal{M}/B_i$  pour tout  $k \in \{1, \dots, n\} \setminus \{i\}$  (d'après le théorème 1.5, page 27). L'hypothèse de récurrence est énoncée comme suit : pour tout  $k \in \{1, \dots, n\} \setminus \{i\}$ , il existe  $j \in \{1, \dots, n\} \setminus \{i\}$  tel que

$$OPT_u(\mathcal{M}/(B_i \cup A_j)) \geq \frac{1}{n-2} ((n-3)u(B') + u(B'_k))$$

Ceci est vrai en particulier pour  $B'_k = \arg \max_{\ell \in \{1, \dots, n\} \setminus \{i\}} u(B'_\ell)$ , donc  $u(B'_k) \geq \frac{1}{n-1}u(B')$ .  
L'inégalité précédente devient

$$OPT_u(\mathcal{M}/(B_i \cup A_j)) \geq \frac{1}{n-2} \left( (n-3)u(B') + \frac{1}{n-1}u(B') \right) = \frac{n-2}{n-1}u(B')$$

Puisque  $u$  est positive, alors  $OPT_u(\mathcal{M}/A_j) \geq OPT_u(\mathcal{M}/(B_i \cup A_j)) + u(B_i) \geq \frac{n-2}{n-1}u(B') + u(B_i) = \frac{n-2}{n-1}u(B \setminus B_i) + u(B_i) = \frac{1}{n-1}((n-2)u(B) + u(B_i))$ . ■

Reprenons la preuve du théorème 1.7. En utilisant le lemme 1.3 en particulier pour  $i = 1$ , on déduit qu'il existe  $j = \sigma(1)$  tel que

$$OPT_u(\mathcal{M}/A_{\sigma(1)}) \geq \frac{1}{n-1}[(n-2)u(B) + u(B_1)] \geq \frac{1}{n-1}[(n-2)OPT_u(\mathcal{M}) + \frac{1}{n}OPT_u(\mathcal{M})]$$

grâce à l'inégalité (1.3). Par conséquent,

$$OPT_u(\mathcal{M}/A_{\sigma(1)}) \geq \frac{n-1}{n}OPT_u(\mathcal{M}) \tag{1.4}$$

Donc le théorème 1.7 est valide pour  $i = 1$ .

En utilisant l'hypothèse de récurrence sur le matroïde  $\mathcal{M}/A_{\sigma(1)}$  et la partition  $\{A_j, j \in \{1, \dots, n\} \setminus \{\sigma(1)\}\}$ , on peut trouver  $\sigma(2), \dots, \sigma(n)$  tels que pour  $i = 2, \dots, n$ ,  $OPT_u(\mathcal{M}/(\cup_{j \leq i} A_{\sigma(j)})) \geq \frac{n-i}{n-1}OPT_u(\mathcal{M}/A_{\sigma(1)}) \geq \frac{n-i}{n-1} \frac{n-1}{n}OPT_u(\mathcal{M}) = \frac{n-i}{n}OPT_u(\mathcal{M})$  grâce à l'inégalité (1.4).

La permutation  $\sigma$  peut être construite en  $O(n^2|X| \max\{\ln |X|, \theta\})$ . En effet, il s'agit de tester chaque indice de  $[n]$ . Soit  $\sigma(1)$  l'indice qui vérifie  $OPT_u(\mathcal{M}/A_{\sigma(1)}) \geq (n-1)/n$ . Nous savons que  $\sigma(1)$  existe grâce au théorème 1.7. Répéter la procédure sur  $\mathcal{M}/A_{\sigma(1)}$  pour trouver  $\sigma(2)$ , c'est-à-dire choisir  $\sigma(2)$  dans  $\{1, \dots, n\} \setminus \{\sigma(1)\}$  qui satisfait  $OPT_u(\mathcal{M}/(A_{\sigma(1)} \cup A_{\sigma(2)})) \geq \frac{n-2}{n-1}OPT_u(\mathcal{M}/A_{\sigma(1)}) \geq \frac{n-2}{n-1} \frac{n-1}{n} = \frac{n-2}{n}$ , et ainsi de suite. ■

### Matroïdes coloriés

Si chaque élément  $e \in X$  a une couleur  $\mathcal{L}(e) \in \{\ell_1, \dots, \ell_p\}$  pour un entier  $p \geq 1$ , alors la structure  $\mathcal{M} = (X, \mathcal{F}, \mathcal{L})$  est dite matroïde colorié. En cas de présence supplémentaire d'une fonction poids  $u : X \rightarrow \mathbb{R}^+$  alors le matroïde  $\mathcal{M} = (X, \mathcal{F}, u, \mathcal{L})$  est dit pondéré colorié.

La définition suivante caractérise la sous-classe des matroïdes coloriés simples :

**Définition 1.17 (Matroïde colorié-simple)** *Un matroïde colorié  $\mathcal{M} = (X, \mathcal{F}, \mathcal{L})$  est dit simple si  $\mathcal{L}(e_1) = \mathcal{L}(e_2)$  pour tout circuit  $\{e_1, e_2\}$  de taille 2.*

Un matroïde simple (voir la définition 1.14, page 26) est colorié-simple. De plus, lorsque chaque couleur apparaît au plus une fois, ces deux notions coïncident.



### 1.3.4 Matroïdes et approximation en multi-objectif

Dans ce qui suit, nous nous intéressons à la version multi-objectif du problème de la base de poids maximum (ou minimum) définie comme suit : soit un matroïde  $\mathcal{M} = (X, \mathcal{F})$  et soit  $\mathcal{B}$  la collection des bases de  $\mathcal{M}$ . On définit  $n \geq 2$  fonctions additives  $u_i : X \rightarrow \mathbb{R}^+$  et  $n$  objectifs où l'objectif  $i$  consiste à trouver une base  $B \in \mathcal{B}$  qui maximise (ou qui minimise)  $u_i(B)$  pour  $i = 1, \dots, n$ .

Ehrgott (1996) s'intéresse aux deux problèmes qui consistent à déterminer l'ensemble de Pareto et le problème de la base  $B$  qui minimise  $\max_{i=1, \dots, n} u_i(B)$ . Les problèmes de décision associés sont NP-complets,  $\forall n \geq 2$  (Camerini et Vercellis, 1984; Warburton, 1985; Ehrgott, 1996). Ehrgott (1996) s'intéresse également à l'adjacence des bases Pareto-optimales : deux bases sont dites adjacentes si elles contiennent  $r(\mathcal{M}) - 1$  éléments en commun. Considérons le graphe dont les sommets sont les bases Pareto-optimales et il existe une arête entre deux bases si et seulement si ces dernières sont adjacentes. Ehrgott (1996) montre que l'ensemble des bases Pareto-optimales supportées<sup>4</sup> est connexe en utilisant la propriété d'échange d'un élément (théorème 1.3, page 27).

Gorski (2010) s'intéresse dans sa thèse à un problème de matroïde bi-objectif où le premier objectif peut prendre des valeurs positives entières mais le second objectif prend des valeurs binaires uniquement. Gorski prouve la connexité de l'ensemble des bases Pareto-optimales et propose un algorithme pour générer l'ensemble de ces bases qui est inspiré de l'algorithme d'intersection de matroïdes.

Nous citons également le problème du sac-à-dos matroïdal (*multi-constrained matroidal knapsack problem*) (Camerini *et al.*, 1989) qui est la version budget du problème multi-objectif de la base de poids maximum (ou minimum). Le problème consiste à trouver une base de poids maximum avec  $p \geq 1$  contraintes de budget (chaque élément a  $p$  différents coûts et chacun des  $p$  coûts de la base sélectionnée doit être inférieur à un budget fixé). Camerini *et al.* (1989) proposent un algorithme basé sur la relaxation Lagrangienne pour calculer des bornes sur la valeur de la solution optimale et qui fournit également une solution approchée avec une garantie probabiliste.

Certains travaux sur les arbres couvrants multi-objectifs (base du matroïde graphique multi-objectif) ont été généralisés aux bases multi-objectifs. Par exemple, le PTAS de Ravi et Goemans (1996) a été généralisé au problème de la base de poids minimum avec une contrainte de budget. Le EPTAS<sup>5</sup> de Hassin et Levin

---

4. Une solution Pareto-optimale est dite supportée si elle appartient à l'enveloppe convexe des solutions Pareto-optimales.

5. EPTAS pour *Efficient Polynomial Time Approximation Scheme* : classe intermédiaire entre PTAS et FPTAS, c'est-à-dire qu'il existe un algorithme qui fournit une  $(1 + \epsilon)$ -approximation pour un problème de minimisation pour tout  $\epsilon > 0$ , de complexité  $O\left(\left(\frac{1}{\epsilon}\right)^{O\left(\frac{1}{\epsilon}\right)}\right)$ .

(2004) est également valide pour ce même problème. Pour  $p \geq 2$  contraintes de budget, Grandoni *et al.* (2009) montrent que le problème admet un PTAS multi-critère en violant les contraintes de budget à un facteur  $(1 + \epsilon)$  pour tout  $\epsilon > 0$ . Si une solution réalisable consiste en un ensemble indépendant (et pas forcément une base) alors Grandoni et Zenklusen (2010) proposent un PTAS.

Les travaux de Berger *et al.* (2011) sur le problème de couplage de poids maximum ont également été généralisés au problème d'intersection de deux matroïdes de poids maximum avec une contrainte de budget où ils montrent que le problème admet un PTAS.

Quant à l'approximation de l'ensemble de Pareto du problème de la base de poids maximum (ou minimum) multi-objectif avec une ou plusieurs solutions, à notre connaissance peu de résultats existent. Il est cependant possible d'utiliser le PTAS du problème de la base de poids minimum avec une contrainte de budget (Ravi et Goemans, 1996) pour approcher l'ensemble de Pareto du problème de la base de poids minimum bi-objectif (voir la section 1.2.2, page 18).

Une question intéressante est l'existence d'un FPTAS pour déterminer une approximation de l'ensemble de Pareto des bases d'un matroïde multi-objectif. Une piste pour avoir un tel algorithme serait de généraliser la routine polynomiale qui résout le problème GAP (problème 14, page 199) de l'arbre couvrant multi-objectif aux bases d'un matroïde multi-objectif. Cela permettrait également d'utiliser le FPTAS proposé par Vassilvitskii et Yannakakis (2005) pour approcher le point d'articulation (la meilleure solution approchée du point idéal).

Dans cette thèse, nous proposons, à notre connaissance, les premiers résultats d'approximation du point idéal du problème de la base de poids maximum multi-objectif (Chapitres 4 et 5). Nous proposons également des résultats d'approximation du point idéal de plusieurs variantes d'un problème bi-objectif défini sur un matroïde pondéré colorié (Chapitre 6).



# Problèmes d'allocation de ressources et notions d'équité

Dans le chapitre 1, il a été mentionné que le problème classique d'allocation de biens indivisibles peut être modélisé par les matroïdes. En raison de notre intérêt pour cette application dans cette thèse, nous présentons dans ce chapitre le cadre général des problèmes d'allocation de ressources et les critères d'équité classiques qui déterminent la qualité d'un partage.

## Sommaire

---

<b>1.1</b>	<b>Théorie de l'approximation en optimisation mono-objectif . . . . .</b>	<b>11</b>
1.1.1	Généralités sur l'approximation . . . . .	12
1.1.2	Conception d'algorithmes d'approximation . . . . .	14
<b>1.2</b>	<b>Approximation en optimisation multi-objectif . . . . .</b>	<b>15</b>
1.2.1	Approche budget ( $\epsilon$ – <i>constraint</i> ) . . . . .	17
1.2.2	Approximation de l'ensemble de Pareto . . . . .	18
1.2.3	Approximation du point idéal . . . . .	21
1.2.4	Inapproximabilité en optimisation multi-objectif . . . . .	23
<b>1.3</b>	<b>Théorie des matroïdes . . . . .</b>	<b>24</b>
1.3.1	Généralités sur les matroïdes . . . . .	24
1.3.2	Exemples de matroïdes . . . . .	27
1.3.3	Matroïdes pondérés . . . . .	31
1.3.4	Matroïdes et approximation en multi-objectif . . . . .	40

---

## 2.1 Problème *générique* d'allocation de ressources

Nous présentons dans cette section un problème *générique* d'allocation de ressources, dit aussi problème MA en référence à Multi-Agents. La ressource est un ensemble  $R = \{e_1, \dots, e_m\}$  de  $m$  éléments. Chaque élément est assimilé à un objet ou à un bien indivisible. Soit  $\mathcal{R} \subseteq 2^R$  une collection de sous-ensembles de  $R$

représentant les portions de la ressource qu'il est possible d'allouer (dites aussi allocations admissibles). Cela signifie que la ressource est soumise à des *contraintes d'admissibilité* qui est un ensemble de conditions imposées sur la ressource, qui restreint l'ensemble des allocations possibles (voir Bouveret (2007) pour une définition formelle). Nous nous intéressons aux ensembles  $S \in \mathcal{R}$  qui satisfont la définition suivante.

**Définition 2.1 (Ensemble maximal pour l'inclusion)** *Un ensemble  $S \in \mathcal{R}$  est dit maximal pour l'inclusion s'il n'existe pas  $S' \in \mathcal{R}$  tel que  $S \subset S'$ .*

Cette hypothèse est communément admise et fait référence au critère de maximalité pour la cardinalité en tant que critère d'efficacité (Bouveret, 2007). Notons  $\mathcal{R}_{\max} \subseteq \mathcal{R}$  la collection des ensembles maximaux pour l'inclusion. Dans ce qui suit, nous nous restreignons aux ensembles  $S \in \mathcal{R}_{\max}$ .

Les données du problème MA comportent aussi un ensemble  $N$  de  $n = |N|$  agents. Une solution réalisable du problème MA, aussi appelée *allocation*, est un vecteur de  $n$  ensembles  $(S_1, \dots, S_n)$  dont l'union  $S = \cup_{i \in N} S_i$  est un ensemble de  $\mathcal{R}_{\max}$ .  $S_i$  est un sous-ensemble de  $R$  et un membre de  $2^R$ , et il représente la part de l'agent  $i \in N$ . Nous supposons que les biens ne sont pas partageables, c'est-à-dire qu'il n'est pas possible d'allouer une même portion de la ressource à différents agents. Ainsi,  $S_i \cap S_j = \emptyset$  pour tout  $i \neq j$ . Cette dernière contrainte est également connue sous le nom de contrainte de *préemption* (Bouveret, 2007). Finalement,  $\{S_1, \dots, S_n\}$  est une *partition* de  $S \in \mathcal{R}_{\max}$ .

Les agents peuvent exprimer des préférences sur les parts admissibles. Ces préférences peuvent être répertoriées en deux classes, selon qu'elles soient *ordinales* ou *cardinales*. Une préférence ordinaire est basée sur une relation binaire sur des portions de la ressource, notée  $\preceq$ , qui est réflexive<sup>1</sup> et transitive<sup>2</sup> (et éventuellement totale<sup>3</sup>) (Chevalerey *et al.*, 2006). Une préférence cardinale peut être représentée par une échelle d'intervalle définie à une transformée affine positive près. Les agents peuvent représenter leurs préférences (ordinales ou cardinales) avec des fonctions d'utilité. Si les préférences sont ordinales, alors la fonction d'utilité est un codage numérique de l'ordre établi et est définie à une transformée strictement croissante près.

Nous supposons dans cette thèse que les préférences sont cardinales et nous définissons la fonction d'utilité  $u_i : 2^R \rightarrow \mathbb{R}^+$  attribuée par l'agent  $i \in N$  (et qui lui est propre) à chaque portion de la ressource, représentant de façon précise son

---

1. Une relation  $\mathcal{S}$  est réflexive si et seulement si  $x\mathcal{S}x$  pour tout  $x$ .  
 2. Une relation  $\mathcal{S}$  est transitive si et seulement si  $x\mathcal{S}y$  et  $y\mathcal{S}z \Rightarrow x\mathcal{S}z$  pour tout triplet  $(x, y, z)$ .  
 3. Une relation  $\mathcal{S}$  est totale si et seulement si  $x\mathcal{S}y$  ou  $y\mathcal{S}x$  pour tout couple  $(x, y)$ .

degré de préférence<sup>4</sup> conformément à l'échelle établie.

En vue du nombre très grand de portions admissibles que peut contenir la ressource  $R$ , il convient de définir des *langages* permettant de restreindre l'énumération des préférences à certaines portions de  $R$ . Par exemple, lorsque la ressource contient  $m$  éléments, on peut utiliser des fonctions d'utilité *additives*. Cela signifie qu'on dispose de l'utilité d'un agent pour tout bien de  $R$  et qu'on définit  $u_i(R')$ , l'utilité de l'agent  $i$  pour la part  $R' \in \mathcal{R}$ , comme étant la somme des utilités de  $i$  pour les éléments de  $R'$ . Pour ne pas alourdir les notations, il est courant de noter  $u_i(e)$  l'utilité de l'agent  $i$  pour  $e \in R$ . On a donc tantôt  $u_i : R \rightarrow \mathbb{R}^+$  pour définir les utilités sur les biens pris séparément, tantôt  $u_i : 2^R \rightarrow \mathbb{R}^+$  pour définir les utilités des agents sur les parts. Dans ce cas, il suffit de connaître les utilités des agents pour les biens individuels. Les agents auront alors à communiquer  $nm$  utilités au lieu de  $n2^m$ . Chevaleyre *et al.* (2006) font un état de l'art sur les langages pour la représentation des préférences.

Dans certains cas, les agents peuvent attribuer des utilités très éloignées aux biens. Par exemple, l'utilité d'un premier agent pour son bien préféré  $e_1$  est 1, tandis que l'utilité d'un deuxième agent pour son bien préféré  $e_2$  est 1000. Cela ne signifie aucunement que le deuxième agent préfère  $e_2$  mille fois plus que le premier agent préfère  $e_1$ , mais que les deux agents ont des échelles différentes pour évaluer les biens. Afin de pallier cette ambiguïté, on peut définir une échelle de grandeur commune pour tous les agents en normalisant l'instance comme suit.

**Définition 2.2 (Instance normalisée à 1)** *Une instance du problème générique d'allocation de ressources est normalisée à 1 si  $\max_{S \in \mathcal{R}} u_i(S) = 1$  pour tout  $i \in N$ . Nous supposons ici que la part de l'agent  $i$  est l'intégralité de la solution  $S$ .*

Nous nous focalisons dans cette thèse sur des collections  $\mathcal{R}$  pour lesquelles l'optimum  $OPT_i = \max_{S \in \mathcal{R}} u_i(S)$  de chaque agent  $i \in N$  se calcule en temps polynomial. Dans ce cas, la normalisation d'une instance peut être réalisée en temps polynomial pour des utilités additives. Il suffit de calculer  $OPT_i$  pour tout  $i \in N$ , puis pour chaque élément  $e \in R$ , diviser l'utilité  $u_i(e)$  par  $OPT_i$  pour le modèle additif, qui est le cas qui nous intéresse dans cette thèse.

Notons que nous nous appuyons sur une hypothèse de monotonie, c'est-à-dire que les agents ont des utilités positives pour tous les éléments. Ainsi, il ne sera jamais négatif pour un agent de recevoir un élément (Bouveret, 2007).

Un agent est un décideur pouvant prendre la forme d'une personne ou d'une entité. Un système multi-agents (SMA) fait référence à un groupe d'agents à qui

---

4. Formellement, il aurait fallu définir  $u_i : 2^R \times \mathcal{R}_{\max} \rightarrow \mathbb{R}^+$  ( $i = 1, \dots, n$ ) car l'utilité d'un agent pour la part qu'il reçoit dépend de la solution qui a été partagée. Mais puisque ce modèle n'est pas traité et pour des raisons de simplicité, nous définissons  $u_i : 2^R \rightarrow \mathbb{R}^+$  ( $i = 1, \dots, n$ ).

la ressource est allouée. Les agents peuvent être des exécutants qui suivent les instructions d'un coordinateur, c'est le cas par exemple de personnes qui héritent des biens attribués par un exécuteur testamentaire. Les agents peuvent également interagir afin de construire ensemble une allocation, comme le ferait une colonie de fourmis pour partager la nourriture récoltée. Pour une description détaillée des SMA, voir par exemple Weiss (2013); Wooldridge (2009).

Nous supposons que les agents sont individualistes, c'est-à-dire que chacun souhaite que l'allocation soit au mieux pour lui, et non au mieux pour les agents en tant que collectivité. Pour un agent, cela revient à considérer sa part de façon absolue, ou relativement aux parts des autres agents. Mais l'objectif est de trouver une solution de compromis qui soit équitable pour ces agents. Nous développerons le concept d'*équité* dans la section 2.2, page 54. Finalement, le problème MA peut être résumé comme suit.

PROBLÈME MA	
<i>Instance :</i>	un ensemble $R$ de $m \geq 1$ éléments, une collection $\mathcal{R} \subseteq 2^R$ , un ensemble $N$ de $n \geq 1$ agents et des utilités $u_i : 2^R \rightarrow \mathbb{R}^+, \forall i \in N$ .
<i>Solution réalisable :</i>	un vecteur de $n$ ensembles deux à deux disjoints $(S_1, \dots, S_n)$ dont l'union $S = \cup_{i \in N} S_i \in \mathcal{R}_{\max} \subseteq \mathcal{R}$ .
<i>Objectif :</i>	satisfaire un certain critère d'équité en tenant compte de $(S_1, \dots, S_n)$ et $(u_1(S_1), \dots, u_n(S_n))$ .

Nous mentionnons dans ce qui suit un lien fort entre l'approche multi-agents (MA) et une autre approche, dite MO, en référence à l'optimisation Multi-Objectif.

En multi-objectif (MO), un décideur prend en compte et optimise différents attributs d'une solution (coût, durée, confort, etc.). Mais derrière chaque attribut, il n'y a pas d'entité (ou d'agent) qui se verrait lésée si l'attribut est négligé par rapport à un autre. À l'inverse, en MA, c'est le point de vue de chaque agent qui donne une dimension pour l'évaluation de la solution. Ainsi, en MO, chaque objectif  $i$  peut être évalué sur la solution  $S \in \mathcal{R}_{\max}$  dans sa globalité et la valeur d'un objectif est une fonction  $u_i : \mathcal{R}_{\max} \rightarrow \mathbb{R}^+$ . La problématique MO est résumée comme suit :

## PROBLÈME MO

*Instance* : un ensemble  $R$  de  $m \geq 1$  éléments, une collection  $\mathcal{R} \subseteq 2^R$ , un entier  $n \geq 1$  et des fonctions objectifs  $u_i : \mathcal{R}_{\max} \rightarrow \mathbb{R}^+$ ,  $i = 1, \dots, n$ .

*Solution*

*réalisable* : un ensemble  $S \in \mathcal{R}_{\max}$ .

*Objectifs* : maximiser  $u_i(S)$ ,  $i = 1, \dots, n$ .

Nous pouvons constater la différence entre les approches MA et MO à travers les exemples suivants.

**Exemple 2.1** Prenons l'exemple énoncé par Ehrgott (2005) dans son introduction. Un individu souhaite acheter une voiture. Il considère un ensemble  $R$  contenant différents modèles de véhicules. Pour faire son choix, cet individu tient compte de trois attributs qui sont le prix, le rendement du moteur et la puissance du véhicule. Il en découle 3 objectifs : minimiser le prix, maximiser le rendement et maximiser la puissance du véhicule. Le but est de sélectionner le meilleur compromis, c'est-à-dire le véhicule qui satisfait au mieux ces trois objectifs. Cependant, on ne s'intéresse pas au degré de satisfaction d'un objectif par rapport à un autre. De plus, chaque objectif est évalué sur le véhicule dans sa globalité. Cela correspond bien à la problématique MO.

**Exemple 2.2** Une association caritative récolte un ensemble  $R$  de jouets. Elle décide de distribuer une partie de ces jouets à différentes familles à l'occasion de la journée de l'enfance. Les sous-ensembles de jouets qu'il est possible de donner sont répertoriés dans la collection  $\mathcal{R}$ . L'objectif est de sélectionner un sous-ensemble de jouets dans  $\mathcal{R}$  et de le partager sur les familles de façon équitable. L'intérêt des familles prime. Ainsi, l'utilité de chaque famille est évaluée sur les jouets qu'elle reçoit. De plus, l'équité du partage doit être pris en compte si les responsables de l'association ne veulent pas avoir de familles lésées. Il s'agit donc d'une problématique MA.

En outre, un point commun entre les problématiques MA et MO est le fait qu'une solution est évaluée à l'aide d'un vecteur. En MO, le nombre d'objectifs est typiquement constant. À l'inverse des problèmes MA, la dimension du vecteur qui évalue une solution est le nombre d'agents, donc il est typiquement non constant.

Notons que la problématique MO peut être étudiée suivant une approche MA. Un cas illustratif sera donné dans l'exemple 2.5 (page 50). D'ailleurs, le problème MO peut être réduit à un problème MA en posant  $u_i(S_i) = u_i(S)$  pour tout  $S_i \subseteq S$ . En outre, dans le contexte qui nous intéresse (les matroïdes avec des



utilités additives d'un point de vue MA), nous prouverons dans la section 2.1.2 (page 49) que le problème MA se réduit au problème MO.

Le problème générique d'allocation de ressources est difficile à résoudre. D'une part, la collection  $\mathcal{R}$  peut contenir un nombre exponentiel d'ensembles. D'autre part, rechercher un sous-ensemble de  $\mathcal{R}$  qui satisfait certaines performances peut être NP-difficile. Quelques exemples de problèmes NP-difficiles seront exposés dans la section 2.2. De ce fait, il serait intéressant d'étudier des collections  $\mathcal{R}$  qui forment des structures particulières.

Un cas particulier intéressant est lorsque  $\mathcal{R} = 2^R$ . Dans ce cas, tout sous-ensemble d'éléments est admissible et l'unique solution réalisable (maximale pour l'inclusion) est  $R$ . Le problème MA consiste alors à allouer la ressource dans son intégralité aux agents. Ce problème est connu dans la littérature sous le nom du problème *classique* de partage ou d'allocation de ressources (biens indivisibles). Notons que le problème présenté dans cette section porte la mention de problème *générique* d'allocation de ressources. Le problème *classique* est présenté dans la section suivante.

### 2.1.1 Problème *classique* d'allocation de ressources

Étant donnée une ressource  $R$  et un ensemble  $N$  de  $n$  agents, l'objectif est de partager la ressource dans son intégralité sur les agents de façon équitable. Il n'existe pas de notion universelle qui caractérise l'équité, mais plusieurs concepts ont été définis dans la littérature que nous exposerons dans la section 2.2.

Le problème classique de partage de ressources se subdivise en deux grandes catégories, selon que la ressource soit divisible ou qu'elle soit constituée d'un ensemble de biens indivisibles.

Dans le cas où la ressource est divisible, le problème est plus connu sous le nom du problème de partage de gâteau (*cake-cutting*). La ressource est assimilée à un gâteau  $R$ , l'aspect divisible s'explique par la possibilité de découper le gâteau en morceaux aussi petits que l'on souhaite et qui seront alloués aux agents. Idéalement, chaque agent souhaiterait s'approprier le gâteau en entier, mais en présence de  $n \geq 2$  agents, le problème consiste à partager équitablement le gâteau sur les  $n$  agents. Une description formelle est définie dans le problème 19, page 200.

Lorsque la ressource est constituée de biens indivisibles, cette dernière correspond à un ensemble  $R = \{e_1, \dots, e_m\}$  de  $m$  biens indivisibles. Pour un bien  $e_j \in R$  et un agent  $i \in N$  donnés, soit  $i$  reçoit  $e_j$  intégralement, soit  $i$  ne reçoit pas  $e_j$ . Une allocation est une partition  $S$  de  $R$  en  $n$  parts  $S = \{S_1, \dots, S_n\}$  (c'est-à-dire,  $\bigcup_{i \in N} S_i = R$  et  $S_i \cap S_j = \emptyset$ ,  $\forall i \neq j$ ) où  $S_i$  est la part de l'agent  $i \in N$ . Tous les

biens sont alors alloués (contrainte de distribution totale). Le problème est résumé comme suit.

PROBLÈME CLASSIQUE D'ALLOCATION DE BIENS INDIVISIBLES	
<i>Instance</i> :	un ensemble $R$ de $m \geq 1$ biens indivisibles, un ensemble $N$ de $n \geq 1$ agents et des utilités $u_i : 2^R \rightarrow \mathbb{R}^+$ , $i = 1, \dots, n$ .
<i>Solution réalisable</i> :	une allocation $S = \{S_1, \dots, S_n\}$ (partition de $R$ ).
<i>Objectif</i> :	satisfaire un certain critère d'équité en tenant compte de $\{S_1, \dots, S_n\}$ et $(u_1(S_1), \dots, u_n(S_n))$ .

Il convient de s'intéresser au *bien-être social* (*social welfare*) pour évaluer la qualité d'une allocation d'un point de vue global, ou en d'autres termes pour évaluer l'équité d'une allocation. Cette évaluation peut se faire au moyen d'un *ordre de bien-être collectif* qui est un préordre<sup>5</sup>  $\succeq$  sur les allocations. Ou encore au moyen d'une *fonction d'utilité collective* qui est une fonction d'agrégation sur le vecteur d'utilité des agents  $(u_i(S_i))_{i \in N}$  pour une allocation  $S = \{S_1, \dots, S_n\}$  de  $R$ . Certaines mesures du bien-être social sont définies dans la Section 2.2.

Pour une description détaillée du problème classique d'allocation de ressources, se référer à Brams et Taylor (1996); Chevaleyre *et al.* (2006). Pour les biens indivisibles en particulier, voir Chevaleyre *et al.* (2006); Bouveret (2007); Lumet (2012).

### 2.1.2 Généralisation du problème classique aux matroïdes

L'objectif de cette thèse est d'étudier une extension ou une classe plus large de collections  $\mathcal{R}$  que celle du problème classique d'allocation de biens indivisibles définie par  $\mathcal{R} = 2^R$ . Nous nous intéressons aux collections  $\mathcal{R}$  qui contiennent les ensembles indépendants d'un matroïde, c'est-à-dire lorsque  $(R, \mathcal{R})$  définit un matroïde. Dans ce cas, les ensembles de  $\mathcal{R}_{\max}$  maximaux pour l'inclusion correspondent aux bases du matroïde  $(R, \mathcal{R})$ . Une telle collection permet donc de généraliser le problème classique d'allocation de biens indivisibles. Remarquons en effet que lorsque  $\mathcal{R} = 2^R$  alors  $(R, \mathcal{R})$  est le matroïde libre (voir page 28). Cette généralisation permet également de capturer des situations plus générales que le modèle classique d'allocation de biens indivisibles. À titre d'exemple, nous mentionnons les applications suivantes.

**Exemple 2.3** Un grand musée national va prêter certaines de ses œuvres d'art à des musées situés dans différentes régions provinciales. Les œuvres du grand

5. Une relation de préordre  $\succeq$  est une relation binaire, réflexive et transitive.

musée national sont réparties selon différentes catégories (statues, peintures, poteries, etc.). Pour chaque catégorie, il existe une borne supérieure sur le nombre d'œuvres que le conservateur du musée principal accepte de prêter. Quelles œuvres d'art choisir et comment les allouer aux musées provinciaux ? Du point de vue des citoyens dont la région accueillera certaines œuvres, l'allocation devrait être équitable. En effet, personne ne veut voir le musée de sa région moins attractif qu'un autre.

L'exemple 2.3 peut être modélisé par le matroïde de partition (voir page 30).  $X$  est le stock du musée national partitionné en  $k$  catégories. Chaque  $b_i$  est le nombre maximum d'œuvres de type  $X_i$  que le conservateur accepte de transférer. Une base contient  $b_i$  œuvres par catégorie  $X_i$ . Ces bases sont évaluées par les agents qui sont les représentants des régions provinciales.

**Exemple 2.4** Un projet de réalisation urbain consiste à réaliser des routes permettant de connecter  $k$  sites. Par souci d'économie, le nombre de routes doit être minimum, c'est-à-dire que deux sites ne peuvent avoir qu'une seule connexion. Le projet sera réalisé par  $n$  différentes entreprises, chacune d'elle a un profit généré sur la réalisation de chaque route. L'objectif est de construire un réseau connexe et répartir sa réalisation sur les  $n$  entreprises de façon à ce que le profit récolté par chacune soit équitable.

L'exemple 2.4 peut être modélisé par le matroïde graphique (voir page 28). Les sommets du graphe sont les  $k$  sites et les arêtes sont les routes qu'il est possible de construire entre deux sites. Une base du matroïde est un arbre couvrant du graphe permettant de connecter les sites de façon minimale. Les arêtes de l'arbre couvrant doivent être partagées sur  $n$  agents qui sont les entreprises de réalisation.

**Exemple 2.5** Un laboratoire est constitué de  $n$  équipes qui ont un séminaire en commun. Le séminaire est composé de  $k$  dates fixes. Il y a  $k$  orateurs qui doivent être sélectionnés parmi un ensemble de candidats. Les candidats ont au préalable donné leurs disponibilités pour ces  $k$  dates. Chaque équipe a son propre intérêt pour les candidats mais l'ordre dans lequel les exposés sont faits n'a pas d'importance. Une solution est un sous-ensemble d'orateurs, sous la contrainte qu'il y ait un orateur disponible par date. Par souci d'équité, il faut également tenir compte de l'intérêt de chaque équipe pour les candidats à sélectionner.

L'exemple 2.5 peut être modélisé par le matroïde transversal (voir page 29).  $X$  est l'ensemble des candidats et  $X_i$  est l'ensemble des candidats disponibles le jour  $i$  pour  $i = 1, \dots, k$ . Une base est un ensemble de  $k$  orateurs avec un orateur disponible par date du séminaire. Les agents sont les  $n$  équipes.

Notons que cet exemple peut être vu comme un problème MO avec une approche MA. En effet, le séminaire dans sa globalité intéressera toutes les équipes qui devront assister à l'ensemble des exposés.

**Exemple 2.6** Le même laboratoire alloue des bourses doctorales à des étudiants sélectionnés parmi un ensemble de candidats  $E$ . Il y a  $k$  types de bourses  $s_1, \dots, s_k$  et au plus  $b_i$  bourses de type  $s_i$  ( $i = 1, \dots, k$ ) peuvent être offertes. Chaque étudiant  $e \in E$  est éligible pour une seule catégorie de bourse  $p(e) \in \{s_1, \dots, s_k\}$ . Les candidats sont choisis par un comité composé des représentants des  $n$  équipes du laboratoire. Chaque équipe  $i \in [n]$  a une utilité positive  $u_i(e)$  pour chaque candidat  $e \in E$ . L'objectif est de sélectionner un sous-ensemble de candidats avec une affectation réalisable en tenant compte de l'intérêt des équipes.

L'exemple 2.6 peut être modélisé par le matroïde de partition (voir page 30). Soit  $X_i = \{e \in E : p(e) = s_i\}$  pour  $i = 1, \dots, k$ . Un ensemble indépendant est un sous-ensemble d'au plus  $b_i$  candidats par  $X_i$ . Les équipes du laboratoire représentent les agents du système.

### Matroïde-allocation

Soit  $\mathcal{M} = (X, \mathcal{F})$  un matroïde et  $N = \{1, \dots, n\}$  un ensemble d'agents. Si l'on s'intéresse au problème MA (voir page 46), alors l'objectif est de trouver un vecteur de  $n$  ensembles deux à deux disjoints  $(B_1, \dots, B_n)$ , dont l'union  $B = \cup_{i \in N} B_i$  est une base du matroïde. Ce problème a deux difficultés, il faut trouver une base et la partitionner en  $n$  parts. Afin de pallier cette double difficulté, nous définissons un nouveau matroïde, appelé le *matroïde-allocation*, où chaque base est déjà allouée aux agents.

On appelle *base-allocation*, une base  $B$  de  $\mathcal{M}$  partitionnée en  $n$  sous-ensembles  $\{B_1, \dots, B_n\}$  où  $B_i$  est la part de l'agent  $i \in N$ . Étant donné un matroïde  $\mathcal{M} = (X, \mathcal{F})$ , on définit dans ce qui suit un autre matroïde  $(Y, \mathcal{E})$ . Le but est de manipuler des bases-allocations. En effet, une base de  $(Y, \mathcal{E})$  est une base-allocation de  $\mathcal{M}$ .

Soit un matroïde  $\mathcal{M} = (X, \mathcal{F})$  où  $X = \{e_1, \dots, e_m\}$ . Pour tout  $i \in [m]$ , soit  $Y_i = \{y_i^1, \dots, y_i^n\}$  où  $y_i^j$  est une copie de  $e_i$  affectée à l'agent  $j$ . Posons  $Y = \cup_{i=1}^m Y_i$ . Pour tout  $E \subseteq Y$ , soit  $p(E)$  l'ensemble  $\{e_i \in X : |E \cap Y_i| > 0, i = 1, \dots, m\}$ ; notons que  $p(E)$  est un ensemble et non une collection. Soit  $\mathcal{E} = \{E \subseteq Y : (p(E) \in \mathcal{F}) \wedge (|E \cap Y_i| \leq 1, i = 1..m)\}$ .

**Lemme 2.1** *Si  $\mathcal{M} = (X, \mathcal{F})$  est un matroïde, alors  $(Y, \mathcal{E})$  est un matroïde.*

**Preuve.** Vérifions les trois axiomes des matroïdes (voir la définition 1.12 page 24).

(i)  $\mathcal{M}$  est un matroïde, donc  $\emptyset \in \mathcal{F}$ . En utilisant  $p(\emptyset) = \emptyset$  et  $|\emptyset \cap Y_i| = 0$  pour tout  $i$ , on obtient  $\emptyset \in \mathcal{E}$ .

(ii) Soient  $E, E'$  tels que  $E \subset E' \subseteq Y$  et  $E' \in \mathcal{E}$ . Cela signifie que  $|E' \cap Y_i| \leq 1$  pour tout  $i \in [m]$ , donc  $|E \cap Y_i| \leq 1$  pour tout  $i \in [m]$ . En utilisant la définition de

$p, p(E)$  est un sous-ensemble de  $p(E')$ . Comme  $E' \in \mathcal{E}$ , nous savons que  $p(E') \in \mathcal{F}$ . Puisque  $\mathcal{M}$  est un matroïde, alors tout sous-ensemble de  $p(E')$ , en particulier  $p(E)$ , est dans  $\mathcal{F}$ .

(iii) Soient  $E$  et  $E'$  deux membres de  $\mathcal{E}$  tels que  $|E| < |E'|$ . Il s'ensuit que  $|p(E)| < |p(E')|$ . Puisque  $p(E)$  et  $p(E')$  appartiennent à  $\mathcal{F}$ , alors il existe forcément  $e_{i^*} \in p(E') \setminus p(E)$  tel que  $p(E) + e_{i^*} \in \mathcal{F}$ , d'après l'axiome (iii) appliqué au matroïde  $\mathcal{M}$ . Soit  $y_{i^*}^{j^*}$  l'unique membre de  $E'$  tel que  $p(\{y_{i^*}^{j^*}\}) = e_{i^*}$ . L'ensemble  $E \cap Y_{i^*}$  doit être vide, sinon  $e_{i^*} \in p(E)$ , qui est une contradiction avec  $e_{i^*} \in p(E') \setminus p(E)$ . D'où,  $|(E + y_{i^*}^{j^*}) \cap Y_{i^*}| = 1$ . En conclusion,  $y_{i^*}^{j^*}$  appartient à  $E' \setminus E$  et  $E + y_{i^*}^{j^*} \in \mathcal{E}$ . ■

Notons que  $(X, \mathcal{F})$  et  $(Y, \mathcal{E})$  ont le même rang.

À titre d'exemple, soit  $G = (V, E)$  un graphe simple et soit  $(X, \mathcal{F})$  le matroïde graphique associé à  $G$ . Construire le multi-graphe  $G' = (V, E')$  tel que  $E'$  contient  $n$  copies de chaque arête  $e \in E$ . Pour toute forêt  $F'$  de  $G'$ , il existe une forêt  $F = p(F')$  de  $G$ .

L'appartenance d'un élément  $y_i^j$  à la solution signifie que  $e_i$  est choisi et est affecté à l'agent  $j$ . L'intérêt du lemme 2.1 est que  $(Y, \mathcal{E})$  apporte plus d'informations sur l'allocation des bases que  $(X, \mathcal{F})$ . De plus, les propriétés du matroïde sont préservées.

Le matroïde  $(Y, \mathcal{E})$  est appelé le *matroïde-allocation* du matroïde  $\mathcal{M} = (X, \mathcal{F})$ .

**Exemple 2.7** Soit  $\mathcal{M} = (X, \mathcal{F})$  un matroïde défini par  $X = \{e_1, e_2, e_3\}$  et  $\mathcal{F} = \{\emptyset, \{e_1\}, \{e_2\}, \{e_3\}, \{e_1, e_3\}, \{e_2, e_3\}\}$ . Les bases sont  $\{e_1, e_3\}$  et  $\{e_2, e_3\}$ . On considère un ensemble  $N = \{1, 2\}$  de deux agents. L'objectif est de trouver une base que les agents de  $N$  partageront.

On construit le matroïde  $(Y, \mathcal{E})$  tel défini précédemment. Ainsi,  $Y = \bigcup_{i=1}^3 Y_i$  où  $Y_i = \{y_i^1, y_i^2\}$  pour  $i = 1, 2, 3$ ,  $p(E) = \{e_i \in X : |E \cap Y_i| > 0, i = 1, 2, 3\}$  et  $\mathcal{E} = \{E \subseteq Y : (p(E) \in \mathcal{F}) \wedge (|E \cap Y_i| \leq 1, i = 1, 2, 3)\}$ .

Par exemple, l'ensemble  $E = \{y_1^1, y_3^2\}$  est une base de  $(Y, \mathcal{E})$ . En effet,  $p(E) = \{e_1, e_3\}$  est une base de  $\mathcal{M}$  et  $|E \cap Y_i| \leq 1$  pour  $i = 1, 2, 3$ . La base-allocation résultante est  $B = \{B_1, B_2\}$  telle que  $B_1 = \{e_1\}$  et  $B_2 = \{e_3\}$  où  $B_i$  est la part attribuée à l'agent  $i \in N$ .

## Notre problème

Nous supposons que les utilités  $u_i$  sont additives, c'est-à-dire que  $u_i : R \rightarrow \mathbb{R}^+$  et  $u_i(R') = \sum_{e \in R'} u_i(e)$  pour tout  $R' \subseteq R$  et pour tout  $i = 1, \dots, n$ .

Nous montrons dans ce qui suit que lorsqu'on considère des collections  $\mathcal{R}$  telles que  $(R, \mathcal{R})$  définit un matroïde et que les utilités sont additives, alors le problème MA se réduit polynomialement au problème MO (voir la section 2.1, page 43).

**Proposition 2.1** *Si  $(R, \mathcal{R})$  définit un matroïde et que les utilités  $(u_i)_{i \in N}$  sont additives alors le problème MA se réduit polynomialement au problème MO.*

**Preuve.** Soit une instance du problème MA définie par l'ensemble  $R$  et la collection  $\mathcal{R}$  de sorte que  $(R, \mathcal{R})$  est un matroïde et soient les fonctions d'utilité additives  $u_i : R \rightarrow \mathbb{R}^+$ ,  $i = 1, \dots, n$ . Nous allons construire une instance du problème MO comme suit.

Soit  $(Y, \mathcal{E})$  le matroïde-allocation de  $(R, \mathcal{R})$  et posons

$$u'_k(y_i^j) = \begin{cases} u_k(e_i) & \text{si } k = j \\ 0, & \text{sinon} \end{cases}$$

pour tout  $k, j \in N$  et pour tout  $i \in [m]$ . Alors pour toute base  $E$  de  $(Y, \mathcal{E})$ , on peut déduire la base  $B = p(E)$  de  $(R, \mathcal{R})$  partitionnée en  $n$  parts  $B = \{B_1, \dots, B_n\}$  où  $B_j = \{e_i \in p(E) : y_i^j \in E\}$ . D'après la définition de  $u'_k$  et puisque cette fonction est additive, on obtient

$$u'_k(E) = \sum_{y_i^j \in E} u'_k(y_i^j) = \sum_{\substack{y_i^j \in E \\ k=j}} u_k(e_i) = \sum_{e \in B_k} u_k(e) = u_k(B_k)$$

car  $u_k$  est additive pour tout  $k \in N$ . Donc  $Y, \mathcal{E}, N, u'_i, i = 1, \dots, n$  est une instance du problème MO. La réduction est évidemment réalisée en temps polynomial. ■

Nous nous concentrons dans cette thèse sur le problème MA lorsque  $(R, \mathcal{R})$  est un matroïde. Finalement, le problème que nous traitons est défini comme suit.

PROBLÈME DE LA BASE D'UN MATROÏDE MULTI-AGENTS	
<i>Instance</i>	un matroïde $\mathcal{M} = (X, \mathcal{F})$ , un ensemble $N$ de $n \geq 1$ agents et $n$ fonctions d'utilité additives $u_i : X \rightarrow \mathbb{R}^+$ , $i = 1, \dots, n$ .
<i>Solution réalisable</i> :	une base-allocation $B = \{B_1, \dots, B_n\}$ de $\mathcal{M}$ .
<i>Objectif</i> :	satisfaire un certain critère d'équité en tenant compte de $\{B_1, \dots, B_n\}$ et $(u_1(B_1), \dots, u_n(B_n))$ .

Rappelons que lorsque  $\mathcal{M}$  est le matroïde libre alors on obtient exactement le problème classique d'allocation de biens indivisibles.

Quant au critère d'équité à appliquer, on peut utiliser l'une des notions définies dans la prochaine section.

## 2.2 Notions de bien-être social et d'équité

L'équité d'une allocation ou d'un partage peut se mesurer par le bien-être social (*social welfare*) pour lequel plusieurs mesures ont été proposées (Brams et Taylor, 1996; Suzumura *et al.*, 2010). Nous présentons dans cette section quelques-unes de ces mesures. Notre but est d'appliquer ces concepts d'équité au problème générique d'allocation de biens indivisibles (problème MA).

Une allocation est un vecteur de  $n \geq 2$  ensembles deux à deux disjoints  $(S_1, \dots, S_n)$ , dont l'union  $S = \cup_{i \in N} S_i$  est un ensemble de  $\mathcal{R}_{\max}$ . Donc  $\{S_1, \dots, S_n\}$  est une partition de  $S$ . Par souci de simplicité, nous noterons  $S = \{S_1, \dots, S_n\} \in \mathcal{R}_{\max}$ . Dans le cas du problème classique,  $S = R$ . Nous supposons dans ce qui suit que les utilités des agents sont additives. Une instance est dite normalisée à 1 si  $\max_{S \in \mathcal{R}_{\max}} u_i(S) = 1$  pour  $i = 1, \dots, n$ . Dans le cas du problème classique, cela signifie que  $u_i(R) = 1$ ,  $i = 1, \dots, n$ .

### Utilitarisme

Le bien-être social utilitariste d'une allocation  $S = \{S_1, \dots, S_n\} \in \mathcal{R}_{\max}$  est défini par  $\sum_{i \in N} u_i(S_i)$ . L'allocation optimale au sens utilitariste est celle qui maximise le bien-être social utilitariste, c'est-à-dire  $\max_{S \in \mathcal{R}_{\max}} \sum_{i \in N} u_i(S_i)$ . On trouve d'autres appellations de ce problème comme l'*optimalité* dans Caragiannis *et al.* (2012), ou encore l'*efficacité* dans Chevaleyre *et al.* (2007a) pour le problème classique.

Ce problème peut être résolu en temps polynomial en appliquant l'algorithme suivant. Pour tout  $e \in R$ , poser  $u(e) = \max_{i \in N} u_i(e)$ . Puis, choisir l'ensemble  $S \in \mathcal{R}$  tel que  $u(S)$  est maximum. Une fois l'ensemble  $S$  choisi, l'allocation est simple à obtenir. Il suffit de poser pour tout  $i \in N$ ,  $S_i = \{e \in S : u(e) = u_i(e)\}$  en prenant soin de vérifier que  $S_i \cap S_j = \emptyset$  pour tout  $i \neq j$ .

### Égalitarisme

Le bien-être social égalitariste d'une allocation  $S = \{S_1, \dots, S_n\} \in \mathcal{R}_{\max}$  est défini par  $\min_{i \in N} u_i(S_i)$ . L'allocation optimale au sens égalitariste est celle qui maximise le bien-être égalitariste, c'est-à-dire  $\max_{S \in \mathcal{R}_{\max}} \min_{i \in N} u_i(S_i)$ .

Le problème de décision associé est  $\text{NP}$ -complet même dans le cas particulier du problème classique et lorsque  $n = 2$  par une réduction de PARTITION (problème 20, page 200).

Dans le cas du problème classique, ce problème est également connu sous le nom *max-min fairness* dans Golovin (2005); Khot et Ponnuswami

(2007); Bezáková et Dani (2005); Asadpour et Saberi (2010), ou encore le problème du *Santa Claus* (Père Noël) dans Bansal et Sviridenko (2006); Asadpour *et al.* (2008). Les biens indivisibles sont assimilés à des cadeaux que le père Noël doit distribuer aux enfants, et il veut rendre l'enfant le plus défavorisé, le plus heureux possible. Notons également l'analogie existante avec le problème d'ordonnancement qui consiste à allouer  $m$  tâches sur  $n$  machines identiques qui fonctionnent en parallèle où chaque tâche  $i$  a une durée d'exécution  $p_i$  ( $i = 1, \dots, m$ ). L'objectif est d'occuper les machines le plus tard possible en maximisant la date de fin d'exécution de la machine qui finit le plus tôt (Deurmeyer *et al.*, 1982; Csirik *et al.*, 1992; Woeginger, 1997). Les tâches représentent les biens indivisibles. Les machines sont les agents. Tous les agents ont les mêmes utilités pour les biens. L'utilité d'un bien ou d'une tâche est son temps d'exécution. Woeginger (1997) montre que dans ce cas (lorsque tous les agents ont les mêmes utilités pour les biens), le problème admet un PTAS mais pas de FPTAS car il est NP-complet au sens fort (Garey et Johnson, 1979).

Quant à l'approximation du problème classique, Bezáková et Dani (2005) montrent que le problème est  $(\frac{1}{m-n-1})$ -approché et qu'il est inapproximable au delà de  $\epsilon \geq \frac{1}{2}$ . Asadpour et Saberi (2010) montrent que le problème est  $\Omega(\frac{1}{\sqrt{n \log^3 n}})$ -approché et Khot et Ponnuswami (2007) montrent qu'il est  $(\frac{1}{2n-1})$ -approché. Le meilleur rapport d'approximation connu est celui de Chakrabarty *et al.* (2009) qui proposent un algorithme  $O(\frac{1}{m^\epsilon})$ -approché pour tout  $\epsilon \in \Omega(\frac{\log \log m}{\log m})$  qui s'exécute en  $m^{O(\frac{1}{\epsilon})}$ . Golovin (2005) propose un algorithme qui fournit une utilité d'au moins  $\frac{OPT}{k}$  (où  $OPT$  est la valeur optimale du problème) à au moins  $\lceil (1 - \frac{1}{k})n \rceil$  agents, pour tout  $k \in \mathbb{N}^*$  donné en entrée. Il considère également le cas particulier où les biens se subdivisent en deux catégories : *grands biens / petits biens* (*big goods / small goods*). Les utilités des petits biens sont parmi  $\{0, 1\}$  et celles des grands biens sont parmi  $\{0, x\}$  où  $x > 1$ . Golovin (2005) montre que dans ce cas, le problème est  $\frac{1}{\sqrt{n}}$ -approché. Khot et Ponnuswami (2007) considèrent le cas où tous les biens ont des utilités de trois types différents  $u_i(e) \in \{0, 1, x\}$  tel que  $x > 1$  pour tout  $i \in N$  et pour tout  $e \in R$ . Ils proposent un algorithme qui est  $\frac{\epsilon}{n}$ -approché qui s'exécute en  $O(m^{O(1)} n^{O(\epsilon)})$ . Bansal et Sviridenko (2006) s'intéressent au cas où les utilités des biens  $u_i(e_j) \in \{0, p_j\}$  pour tout  $i \in N$  et pour  $j = 1, \dots, m$  et montrent que le problème est  $O(\frac{\log \log \log n}{\log \log n})$ -approché. Ferraioli *et al.* (2014) traitent le cas *régulier* qui signifie que chaque agent doit recevoir exactement  $k$  biens pour un entier  $k$  donné. Ils montrent que



ce problème est  $\text{NP}$ -complet au sens fort pour tout  $k \geq 2$  et qu'il est  $\frac{1}{k}$ -approximable. De plus, lorsque les utilités des agents sont de deux types  $v_1$  ou  $v_2$  avec  $v_2 > v_1 \geq 0$  alors ce problème peut être résolu en temps polynomial en utilisant l'intersection de deux matroïdes. Lorsque les utilités sont de trois types  $v_1, v_2$  et  $v_3$  avec  $0 \leq v_1 < v_2 < v_3$  alors le problème est  $\rho$ -approché où  $\rho = \frac{v_2 + (k-1)v_1}{kv_2}$  si  $2v_2 \leq v_1 + v_3$  et  $\rho = \frac{v_1 + (k-1)v_2}{v_1 + (k-1)v_3}$  sinon (Ferraioli *et al.*, 2014).

### Juste part max-min

La juste part max-min (*max-min fair share*) d'un agent  $i \in N$  est définie par la valeur  $u_i^{MFS} = \max_{S \in \mathcal{R}_{\max}} \min_{j \in N} u_j(S_j)$ . Il s'agit de l'utilité maximum que peut avoir l'agent  $i$  pour la pire part d'une allocation si tous les agents avaient les mêmes utilités que lui. Une allocation  $S = \{S_1, \dots, S_n\} \in \mathcal{R}_{\max}$  vérifie la juste part max-min si  $u_i(S_i) \geq u_i^{MFS}$  pour tout  $i \in N$ , c'est-à-dire s'il existe une allocation où chaque agent  $i \in N$  a au moins sa juste part max-min  $u_i^{MFS}$ .

La juste part max-min a été introduite récemment par Budish (2011) pour le problème classique. Procaccia et Wang (2014) montrent, contre toute intuition, qu'il existe des instances du problème classique qui n'admettent pas d'allocation qui satisfasse la juste part max-min. De plus, Bouveret et Lemaître (2014) montrent que le problème de décision associé au calcul du vecteur  $(u_i^{MFS})_{i \in N}$  est  $\text{NP}$ -complet. Quant à la complexité exacte du problème d'existence d'une allocation qui satisfait la juste part max-min, le problème reste ouvert. Il est clair que le problème de décision associé n'est pas dans  $\text{NP}$  puisqu'il est  $\text{NP}$ -difficile de calculer le vecteur  $(u_i^{MFS})_{i \in N}$ , par contre il est dans  $\Sigma_2^p$  (Bouveret et Lemaître, 2014). Procaccia et Wang (2014) montrent également une  $\left(\frac{2}{3}\right)_{i \in N}$ -approximation pour le problème classique lorsque le nombre d'agents  $|N| = n$  est constant.

### Proportionnalité

Soit  $OPT_i = \max_{S \in \mathcal{R}_{\max}} u_i(S)$  l'utilité maximum que l'agent  $i \in N$  peut avoir. Une allocation  $S = \{S_1, \dots, S_n\} \in \mathcal{R}_{\max}$  est dite proportionnelle si  $u_i(S_i) \geq \frac{OPT_i}{n}$  pour tout agent  $i \in N$ . Dans le cas du problème classique,  $OPT_i = u_i(R)$ ,  $i \in N$ . C'est le critère originel de partage équitable proposé initialement par Steinhaus (1948). Le principe, assez

---

6.  $\Sigma_2^p = \text{NP}^{\text{NP}}$ . Il s'agit de la classe des problèmes qui peuvent être résolus en temps non-déterministe polynomial par une machine de Turing augmentée d'un oracle  $\text{NP}$ .

naturel, consiste à attribuer au moins  $1/n^{\text{ème}}$  de la ressource à chacun des  $n$  agents dans le cas du problème classique.

Une allocation proportionnelle n'existe pas toujours, comme dans les instances suivantes du problème classique :

**Exemple 2.8**  $R = \{e_1\}$  et  $n = 2$  agents tels que  $u_1(e_1) = u_2(e_1) = 1$ .

En allouant l'unique bien indivisible  $e_1$  à l'un des deux agents, l'allocation n'est pas proportionnelle puisque l'agent qui possède ce bien aura une utilité de 1, tandis que l'utilité de l'agent qui ne l'aura pas est 0.

**Exemple 2.9**  $R = \{e_1, e_2, e_3\}$  et  $n = 2$  agents avec  $u_1(e_1) = 0.6$ ,  $u_1(e_2) = u_1(e_3) = 0.2$ ,  $u_2(e_1) = 0.7$ ,  $u_2(e_2) = 0.2$  et  $u_2(e_3) = 0.1$ .

Il n'existe pas d'allocation qui fournisse à chaque agent une utilité d'au moins 0.5.

Demko et Hill (1988); Bouveret et Lemaître (2014) montrent pour le problème classique que décider de l'existence d'une allocation proportionnelle est  $\mathbb{NP}$ -complet par une réduction de PARTITION (problème 20, page 200). De plus, Markakis et Psomas (2011) montrent que pour toute constante  $c \geq 1$ , il est  $\mathbb{NP}$ -complet de décider de l'existence d'une allocation où l'utilité de chaque agent  $i$  pour sa part vaut au moins  $\frac{u_i(R)}{cn}$  pour tout  $i \in N$ .

Comme alternative à la proportionnalité, on peut chercher une allocation avec une garantie inférieure à  $\frac{u_i(R)}{n}$  pour chaque agent  $i \in N$ . Markakis et Psomas (2011) considèrent le problème classique avec des instances normalisées à 1 et proposent un algorithme polynomial qui construit une allocation pour laquelle l'utilité de chaque agent  $i$  pour sa part vaut au moins  $V_n(\alpha_i) \in \left[0, \frac{1}{n}\right]$  où  $\alpha_i = \max_{e \in R} u_i(e)$  et  $V_n$  est une fonction de  $\alpha_i$  proposée initialement par Hill (1987). La fonction  $V_n$  sera définie dans la section 2.3.

### Absence d'envie

Une allocation  $S \in \mathcal{R}_{\max}$  est sans envie si  $u_i(S_i) \geq u_i(S_j)$  pour tout  $i, j \in N$ . Le but est de trouver une allocation où aucun agent n'est envieux de la part d'un autre. Notons qu'il n'est pas nécessaire que l'instance soit normalisée ici.

Dans le cas du problème classique d'allocation de biens indivisibles (et divisibles) et pour des utilités additives, il est bien connu qu'une allocation sans envie est proportionnelle mais la réciproque est vraie uniquement lorsque  $n = 2$ . L'additivité des fonctions d'utilité est une

condition nécessaire pour vérifier cette propriété, comme on peut le constater dans l'exemple suivant :

**Exemple 2.10** Soit une instance du problème classique d'allocation de biens indivisibles définie par  $R = \{e_1, e_2\}$  avec  $u_i(e_j) = 0.4$  pour tout  $i, j \in \{1, 2\}$  et  $u_i(\{e_1, e_2\}) = 1$  pour  $i = 1, 2$ . Ainsi,  $OPT_i = u_i(\{e_1, e_2\}) = 1$  pour  $i = 1, 2$ . En allouant  $e_i$  à l'agent  $i$  ( $i = 1, 2$ ), on obtient bien un partage sans envie car  $u_i(e_i) = u_i(e_{3-i})$ . Mais ce partage n'est pas proportionnel puisque  $u_i(e_i) < \frac{1}{2}$ ,  $i = 1, 2$ .

Cependant, dans le cas du problème générique d'allocation de biens indivisibles, la propriété précédente n'est pas forcément vérifiée, même si les utilités des agents sont additives.

**Exemple 2.11** Considérons l'instance suivante du problème générique d'allocation de biens indivisibles :

$R = \{e_1, e_2, e_3\}$  et  $\mathcal{R} = \{\{e_1, e_2\}, \{e_1, e_3\}, \{e_2, e_3\}\}$ . Il y a deux agents, leurs utilités sont additives et sont représentées dans le tableau 2.1.

$e$	$e_1$	$e_2$	$e_3$
$u_1(e)$	0.6	0.4	0
$u_2(e)$	0.6	0	0.4

Tableau 2.1 – Exemple 2.11.

Les allocations admissibles qui maximisent les utilités des agents 1 et 2 sont respectivement  $\{e_1, e_2\}$  et  $\{e_1, e_3\}$ . Ainsi,  $OPT_1 = u_1(\{e_1, e_2\}) = 1$  et  $OPT_2 = u_2(\{e_1, e_3\}) = 1$ . L'allocation  $S = \{S_1, S_2\}$  où  $S_1 = \{e_2\}$  et  $S_2 = \{e_3\}$  est sans envie car  $u_1(S_1) = 0.4 > u_1(S_2) = 0$  et  $u_2(S_2) = 0.4 > u_2(S_1) = 0$ . Mais elle n'est pas proportionnelle puisque  $u_i(S_i) < \frac{1}{2}$ ,  $i = 1, 2$ .

Pour le problème de la base d'un matroïde multi-agents, nous proposerons dans la section 3.5 (page 88) une définition de l'absence d'envie plus appropriée qui permettra de vérifier la propriété « absence d'envie  $\Rightarrow$  proportionnalité ; et proportionnalité  $\Rightarrow$  absence d'envie si  $n = 2$  ».

Une allocation sans envie n'existe pas toujours (voir les exemples 2.8 et 2.9, page 57). De plus, il est NP-complet de décider de l'existence d'une telle allocation même dans le cas du problème classique Lipton *et al.* (2004).

Au lieu de chercher une allocation sans envie, on peut s'intéresser à une allocation d'envie minimum. Ce problème a été introduit par Lipton *et al.* (2004) qui montrent que résoudre et approcher le problème classique est NP-difficile.

Lipton *et al.* (2004) introduisent également une autre mesure de l'envie qui est le ratio d'envie minimum  $\max_{i,j} \left\{ 1, \frac{u_i(S_j)}{u_i(S_i)} \right\}$ . Ils montrent que résoudre et approcher le problème d'obtention d'une allocation avec un ratio d'envie minimum est NP-difficile pour le problème classique. Lorsque les agents ont tous les mêmes utilités pour les biens, ils proposent un PTAS en général et un FPTAS pour un nombre constant d'agents. Nguyen et Rothe (2013b) considèrent également le ratio d'envie  $\frac{u_i(S_j)}{u_i(S_i)}$ . Ils définissent le degré d'envie d'un agent  $i$  soit par le ratio maximum  $\max_{j \neq i} \frac{u_i(S_j)}{u_i(S_i)}$  ou par le produit des ratios  $\prod_{j \neq i} \frac{u_i(S_j)}{u_i(S_i)}$ . Le degré d'envie d'une société est aussi défini soit par le maximum ou le produit des degrés d'envie de tous les agents (défini par le maximum ou le produit). Ils proposent un FPTAS pour minimiser le degré d'envie de la société selon les quatre définitions proposées lorsque le nombre d'agents est borné.

### Équitabilité

Supposons que toutes les instances sont normalisées à 1. Une allocation  $S = \{S_1, \dots, S_n\} \in \mathcal{R}$  est dite équitale selon Caragiannis *et al.* (2012); Brams et Taylor (1996) si  $u_i(S_i) = u_j(S_j)$  pour tout couple d'agents  $i \neq j$ . Ce critère signifie que tous les agents doivent recevoir exactement la même quantité selon leurs utilités. Ici encore, une telle allocation n'existe pas toujours (comme dans l'exemple 2.8). De plus, décider de l'existence d'une allocation équitale est NP-complet même pour le problème classique et lorsque  $n = 2$  agents par une réduction du problème PARTITION (problème 20, page 200) que nous exposerons dans la section 3.2 du chapitre 3.

Notons qu'il est inutile de se restreindre aux ensembles de  $\mathcal{R}$  maximaux pour l'inclusion. Mais dans ce cas, l'ensemble vide peut être une solution réalisable puisque  $u_i(\emptyset) = 0$  pour tout  $i \in N$  (si  $\emptyset \in \mathcal{R}$ ). Il faudrait alors s'assurer d'allouer des portions non nulles aux agents.

### Absence de jalousie

Nous proposons la notion de jalousie définie comme suit pour des instances normalisées à 1. Un agent  $i$  n'est pas jaloux d'un autre agent  $j$  dans une allocation  $S = \{S_1, \dots, S_n\} \in \mathcal{R}_{\max}$  si  $u_i(S_i) \geq u_j(S_j)$ . Cela

signifie qu'un agent est jaloux d'un autre si ce dernier jouit d'une plus grande utilité. Un agent  $i$  est sans jalousie s'il n'est jaloux d'aucun autre agent  $j \neq i$ . Une allocation est sans jalousie si aucun agent n'est jaloux. Une allocation sans jalousie correspond à une allocation équitable dans une instance normalisée. L'absence de jalousie sera reprise en détail dans la section 3.3, page 81.

### Pareto-optimalité

Une allocation  $S = \{S_1, \dots, S_n\} \in \mathcal{R}_{\max}$  est Pareto-optimale s'il n'existe pas d'autre allocation  $S' = \{S'_1, \dots, S'_n\} \in \mathcal{R}_{\max}$  telle que  $u_i(S'_i) \geq u_i(S_i)$  pour tout  $i \in N$  et il existe  $j \in N$  tel que  $u_j(S'_j) > u_j(S_j)$ . Donc aucun agent ne peut enrichir sa part sans dégrader la part d'un autre.

D'un point de vue MO, il est plus intéressant que l'allocation choisie soit Pareto-optimale, d'autant plus qu'une telle solution existe toujours. En effet, pour obtenir une allocation Pareto-optimale, il suffit par exemple de choisir un ensemble  $S \in \mathcal{R}_{\max}$  qui maximise l'utilité d'un seul agent et le lui allouer intégralement. Mais d'un point de vue MA, cette solution n'est pas juste par rapport aux autres agents.

Quant à déterminer toutes les solutions Pareto-optimales, le problème est intraitable. Par exemple, considérons une instance du problème classique définie par  $n \geq 2$  agents,  $R = \{e_1, \dots, e_m\}$  et  $u_i(e_j) = 2^j$  pour  $i = 1, \dots, n$  et pour  $j = 1, \dots, m$ . Toute solution est Pareto-optimale, or il existe  $n^m$  solutions. En effet, il y a  $n$  façons d'allouer un seul bien (en l'attribuant à l'un des  $n$  agents) et il y a  $m$  biens à allouer. Quant à vérifier si une allocation donnée est Pareto-optimale, le problème classique est coNP-complet (Bouveret et Lang, 2008).

L'utilitarisme implique la Pareto-optimalité mais le contraire n'est pas vrai. Considérons l'instance suivante du problème classique : soit  $n = 2$  agents et  $R = \{e_1, e_2\}$  avec  $u_1(e_1) = u_1(e_2) = 0.5$ ,  $u_2(e_1) = 0.6$  et  $u_2(e_2) = 0.4$ . L'allocation  $S = \{S_1, S_2\}$  définie par  $S_1 = \{e_1, e_2\}$  et  $S_2 = \emptyset$  est Pareto-optimale mais elle n'est pas optimale au sens utilitariste car il existe une autre allocation  $S' = \{S'_1, S'_2\}$  où  $S'_1 = \{e_2\}$  et  $S'_2 = \{e_1\}$  dont le bien-être utilitariste vaut  $\sum_{i=1,2} u_i(S'_i) = 1.1 > \sum_{i=1,2} u_i(S_i) = 1$ .

Certains travaux traitent de l'association de la Pareto-optimalité à une autre notion d'équité. Par exemple, de Keijzer *et al.* (2009) montrent qu'il est  $\Sigma_2^P$ -complet de décider s'il existe une allocation qui est à la fois sans envie et Pareto-optimale. Cela reste NP-complet même si les utilités sont identiques pour tous les agents, ou encore si les utilités des éléments sont binaires (0 ou 1).

### Dictateur de rang $k$

Étant donnée une allocation  $S = \{S_1, \dots, S_n\} \in \mathcal{R}_{\max}$ , soit  $\sigma$  une permutation de  $\{1, \dots, n\}$  telle que  $u_{\sigma(i)}(S_{\sigma(i)}) \leq u_{\sigma(i+1)}(S_{\sigma(i+1)})$  pour  $i = 1, \dots, n-1$ . Le bien-être social *dictateur de rang  $k$*  pour un certain  $k \in \{1, \dots, n\}$  est  $u_{\sigma(k)}(S_{\sigma(k)})$ , c'est-à-dire l'utilité du  $k^{\text{ème}}$  agent (pour sa part) le moins heureux que l'on cherche à maximiser (Chevaleyre *et al.*, 2006). L'égalitarisme est un cas particulier correspondant à  $k = 1$ . Lorsque  $k = n$ , le modèle est dit élitiste et lorsque  $k = \lceil \frac{n}{2} \rceil$ , il est dit dictateur de rang médian.

### Opérateur OWA

La *moyenne ordonnée pondérée* (OWA pour *ordered weighted averaging*) est une fonction d'agrégation couramment utilisée en optimisation multi-objectif (voir Perny *et al.* (2006); Galand (2008)) qui associe un poids à l'utilité de chaque agent (pour sa part), selon l'ordre des performances de ces utilités. Soit  $\sigma$  une permutation de  $\{1, \dots, n\}$  telle que  $u_{\sigma(i)}(S_{\sigma(i)}) \geq u_{\sigma(i+1)}(S_{\sigma(i+1)})$  pour  $i = 1, \dots, n-1$ . Étant donné un vecteur poids  $w \in \mathbb{R}^n$  tel que  $\sum_{i \in N} w_i = 1$ , la moyenne ordonnée pondérée d'une allocation  $S = \{S_1, \dots, S_n\} \in \mathcal{R}_{\max}$  est définie par  $\sum_{i \in N} w_i u_{\sigma(i)}(S_{\sigma(i)})$ . Il est intéressant d'utiliser l'opérateur OWA comme critère d'équité car il permet de contrôler l'importance attribuée à chaque degré de richesse des agents (principe de Pigou-Dalton...) Weymark (1981); Shorrocks (1983). L'objectif est de trouver une allocation qui maximise cet opérateur.

Si  $w_k = 1$  pour un certain  $k \in N$  et  $w_{k'} = 0$  pour tout  $k' \neq k$  alors l'opérateur OWA correspond au dictateur de rang  $n - k + 1$ , généralisant de ce fait les critères élitiste et égalitariste. Lorsque  $w = (\frac{1}{n}, \dots, \frac{1}{n})$ , l'opérateur OWA revient à résoudre le critère utilitariste.

### Produit de Nash

Le produit de Nash d'une allocation  $S = \{S_1, \dots, S_n\} \in \mathcal{R}_{\max}$  est défini par  $\prod_{i \in N} u_i(S_i)$ . L'optimalité au sens de Nash consiste à maximiser le produit de Nash, c'est-à-dire  $\max_{S \in \mathcal{S}} \prod_{i \in N} u_i(S_i)$ . Ce critère a été introduit par Nash (1950). Selon Nguyen (2013), ce critère représente un compromis entre l'utilitarisme et l'égalitarisme. En effet, si l'utilité d'un agent pour sa part augmente, cela fait augmenter le produit de Nash comme pour le critère utilitariste. De plus, si l'écart entre les utilités des agents pour leurs parts diminue, cela fait généralement augmenter le produit de Nash, tout comme le critère égalitariste qui

visé à maximiser l'utilité de l'agent le moins heureux et par conséquent, réduire l'écart entre les parts des agents.

Nguyen et Rothe (2013a) s'intéressent à maximiser la racine  $n^{\text{ème}}$  du produit de Nash (*the average Nash product*) pour le problème classique. Ils montrent une  $(\frac{1}{m-n+1})$ -approximation basée sur un algorithme polynomial. De plus, lorsque tous les agents ont la même utilité, le problème admet un PTAS.

### Ordre lexicographique minimum

Cette notion vient affiner le critère égalitariste. L'ordre lexicographique minimum (appelé aussi *leximin*) permet de comparer deux allocations  $S, S' \in \mathcal{R}_{\max}$  en comparant d'abord l'agent le moins heureux de chaque allocation. L'emporte l'allocation dont l'utilité de l'agent le moins heureux est la plus grande. En cas d'égalité, comparer entre le deuxième agent le moins heureux de chaque allocation et ainsi de suite. Formellement, soient  $\sigma$  et  $\sigma'$  des permutations de  $\{1, \dots, n\}$  telles que  $u_{\sigma(i)}(S_{\sigma(i)}) \leq u_{\sigma(i+1)}(S_{\sigma(i+1)})$  et  $u_{\sigma'(i)}(S'_{\sigma'(i)}) \leq u_{\sigma'(i+1)}(S'_{\sigma'(i+1)})$  pour  $i = 1, \dots, n-1$ . Alors l'allocation  $S$  est leximin préférée à  $S'$  si et seulement s'il existe  $k \in \{1, \dots, n\}$  tel que  $u_{\sigma(i)}(S_{\sigma(i)}) = u_{\sigma'(i)}(S'_{\sigma'(i)})$  pour tout  $i < k$  et  $u_{\sigma(k)}(S_{\sigma(k)}) > u_{\sigma'(k)}(S'_{\sigma'(k)})$ .

L'objectif est de trouver une allocation qui n'est préférée lexicographiquement par aucune autre.

Si l'instance est normalisée à 1, alors l'optimum selon l'ordre lexicographique minimum est plus connu sous le nom de solution de *Kalai-Smorodinsky* (Moulin, 2004).

Pour une description détaillée des résultats d'approximabilité de ces approches pour le problème classique, nous citons l'état de l'art de Nguyen *et al.* (2013). Pour une approche critique sur les notions d'équité et leur intérêt pour les agents et la société, nous renvoyons le lecteur à Bouveret (2007); Brams et Taylor (1996); Young (1995). Il existe encore plusieurs autres notions de bien-être social développées en choix social décrites par exemple par Suzumura *et al.* (2010) et qui peuvent faire office de critères d'équité (Endriss et Maudet, 2004).

## 2.3 Garantie dans le pire cas

Une vision différente de l'équité que celles proposées dans la section précédente est l'approche de Demko et Hill (1988); Markakis et Psomas (2011). Ces derniers s'intéressent au problème classique d'allocation de biens indivisibles et

supposent que les instances sont normalisées à 1, c'est-à-dire  $u_i(R) = 1$  pour tout  $i \in N$ . Demko et Hill (1988) montrent l'existence d'une allocation avec une garantie dans le pire cas (ou borne inférieure) sur l'utilité de l'agent le moins heureux. Markakis et Psomas (2011) améliorent ce résultat et proposent un algorithme polynomial qui construit une allocation avec une garantie dans le pire cas sur l'utilité de chaque agent. Ces garanties ne sont pas constantes. La garantie de Demko et Hill (1988) dépend d'un paramètre noté  $\alpha = \max_{i \in N} \max_{e \in R} u_i(e)$  représentant l'utilité maximum d'un bien indivisible. La garantie de Markakis et Psomas (2011) dépend des paramètres  $\alpha_i = \max_{e \in R} u_i(e)$ ,  $i = 1, \dots, n$  représentant l'utilité maximum de l'agent  $i$  pour un bien indivisible. Puisque l'instance est normalisée à 1, alors les paramètres  $\alpha$  et  $\alpha_i$  prennent leurs valeurs dans  $[0, 1]$ . Les garanties de Demko et Hill (1988); Markakis et Psomas (2011) sont définies par une famille de fonctions qui dépend respectivement de  $\alpha$  et  $\alpha_i$ . Ces fonctions sont notées  $V_n : [0, 1] \rightarrow [0, \frac{1}{n}]$  où  $n$  est le nombre d'agents.

### 2.3.1 Les fonctions $V_n$

Les fonctions  $V_n$  ont une importance particulière dans cette thèse. En effet, nous proposons dans les chapitres 4 et 5 des garanties qui en dépendent. La famille de fonctions  $V_n$  a été introduite initialement par Hill (1987) pour des mesures de probabilité et est définie comme suit.

**Définition 2.3** *Hill (1987); Markakis et Psomas (2011) Soit un entier  $n \geq 2$ , et soit  $V_n : [0, 1] \rightarrow [0, n^{-1}]$  l'unique fonction décroissante vérifiant  $V_n(x) = 1/n$  pour  $x = 0$ , et pour  $x > 0$  :*

$$V_n(x) = \begin{cases} 1 - p(n-1)x, & x \in I(n, p) \\ 1 - \frac{(p+1)(n-1)}{(p+1)^{n-1}}, & x \in NI(n, p) \end{cases}$$

où  $\forall p \geq 1$  entier,  $I(n, p) = [\frac{p+1}{p((p+1)^{n-1})}, \frac{1}{pn-1}]$  et  $NI(n, p) = ]\frac{1}{(p+1)^{n-1}}, \frac{p+1}{p((p+1)^{n-1})} [$ . Nous ajoutons à cette définition  $V_1(x) = 1$  pour tout  $x \in [0, 1]$ .

La figure 2.1 contient une représentation graphique de  $V_n(x)$  pour  $n = 2$  et  $n = 3$ .

Les fonctions  $V_n$  sont décroissantes et sont définies par des segments délimités par les intervalles  $I(n, p)$  et  $NI(n, p)$  pour  $n \geq 2$  et  $p \geq 1$ . Les segments de  $I(n, p)$  sont strictement décroissants et ceux de  $NI(n, p)$  sont constants. Par exemple, pour  $n = 2$  et  $p = 1$ , on a  $I(2, 1) = [\frac{2}{3}, 1]$  et la fonction  $V_2(x) = 1 - x$  est strictement décroissante. Pour  $NI(2, 1) = ]\frac{1}{3}, \frac{2}{3} [$ , la fonction  $V_2(x) = \frac{1}{3}$  est constante.

Demko et Hill (1988) montrent l'existence d'une allocation où l'utilité de l'agent le moins heureux pour sa part vaut au moins  $V_n(\alpha) \in [0, \frac{1}{n}]$  où  $\alpha = \max_{i \in N} \max_{e \in R} u_i(e)$ .



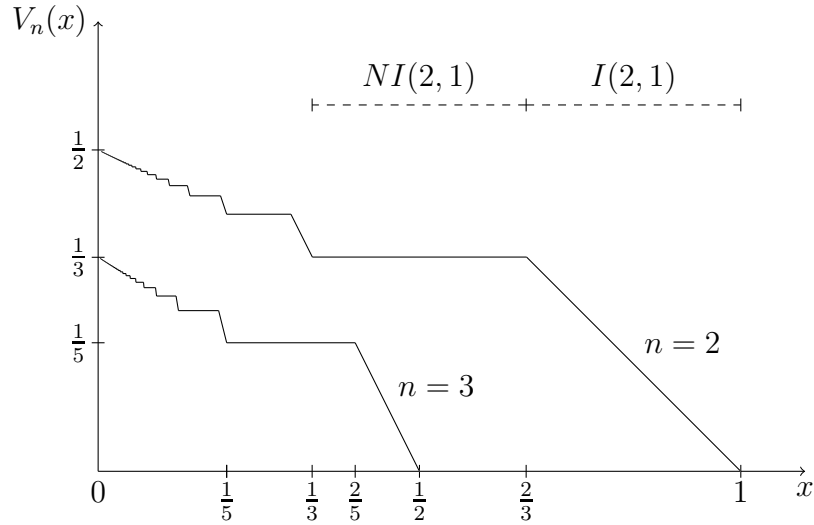


FIGURE 2.1 –  $V_n$  pour  $n = 2$  et  $n = 3$ .

Remarquons que ce résultat n'est pas comparable à l'approche égalitariste. L'approche égalitariste consiste à maximiser l'utilité de l'agent le moins heureux pour des instances pas forcément normalisées. Ou du moins offrir une approximation de cette valeur, c'est-à-dire chercher une borne inférieure du rapport  $\frac{u_i(S_i)}{OPT}$  où  $S = \{S_1, \dots, S_n\}$  est la solution approchée,  $i$  l'agent le moins heureux par rapport à l'allocation  $S$  et  $OPT$  la valeur optimale selon le critère égalitariste. Par contre, Demko et Hill (1988) s'intéressent à borner inférieurement le rapport  $\frac{u_i(S_i)}{u_i(R)}$  où  $R$  est l'ensemble des biens indivisibles. Le rapport  $\frac{u_i(S_i)}{u_i(R)}$  permet de normaliser l'instance à 1 et compare l'utilité de l'agent  $i$  dans l'allocation  $S$  par rapport à sa meilleure utilité s'il prenait tous les biens indivisibles. En conclusion, l'approche égalitariste consiste à étudier le ratio de l'utilité de chaque agent par rapport à l'utilité qu'aurait, dans le meilleur des cas, l'agent le plus défavorisé. Alors que l'approche de Demko et Hill (1988) (garantie dans le pire cas) vise à mesurer l'utilité de la part de chaque agent par rapport à ce qu'il aurait s'il était seul.

Par la suite, Markakis et Psomas (2011) améliorent le résultat de Demko et Hill (1988) en proposant un algorithme polynomial appelé **ALLOCATE** qui construit une allocation pour laquelle l'utilité de chaque agent  $i$  pour sa part vaut au moins  $V_n(\alpha_i) \in \left[0, \frac{1}{n}\right]$  où  $\alpha_i = \max_{e \in R} u_i(e)$ . Puisque  $\alpha \geq \alpha_i$  pour tout  $i \in \{1, \dots, n\}$  et que la fonction  $V_n$  est décroissante alors  $V_n(\alpha_i) \geq V_n(\alpha)$  pour tout  $i \in \{1, \dots, n\}$ . Cette approche est équivalente à approcher le point idéal du problème multi-objectif qui consiste à trouver une allocation des biens indivisibles sur les  $n$  agents et de sorte que chaque objectif  $i \in \{1, \dots, n\}$  consiste à maximiser l'utilité de l'agent  $i$ . En effet, rappelons que lorsque  $(R, \mathcal{R})$  est un matroïde alors le problème MA, qui généralise

le problème classique de partage de biens indivisibles, se réduit au problème MO.

L'algorithme ALLOCATE de Markakis et Psomas (2011) est présenté dans l'algorithme 2.1. Il prend en entrée les biens indivisibles, les agents et leurs utilités pour les biens. Dans cet algorithme, on sélectionne l'agent  $i \in N$  qui peut atteindre la garantie  $V_n(\alpha_i)$  avec le plus petit nombre de biens. Cet agent se voit allouer les biens qu'il a sélectionnés. Puis, cet agent ainsi que ses biens sont supprimés de l'instance. L'algorithme est relancé à chaque fois jusqu'à ce qu'il reste deux agents. À la fin, l'agent qui n'a pas été sélectionné prendra tous les éléments restants.

---

**Algorithme 2.1 : ALLOCATE** (Markakis et Psomas, 2011)

---

**Données :** un ensemble de biens indivisibles  $R$ , un ensemble d'agents  $N$  et des utilités  $(u_i)_{i \in N}$

**Résultat :** une allocation des biens indivisibles où l'utilité de chaque agent  $i \in N$  pour sa part vaut au moins  $V_n(\alpha_i)$

```

1 pour tout  $i \in N$  faire
2    $S_i \leftarrow \emptyset$ 
3 tant que  $\nexists i \in N$  tel que  $u_i(S_i) \geq V_n(\alpha_i)$  faire
4   pour tout  $i \in N$  faire
5      $S_i \leftarrow S_i \cup \{\text{prochain élément d'utilité maximum pour l'agent } i\}$ 
6 choisir un agent  $i \in N$  avec  $u_i(S_i) \geq V_n(\alpha_i)$ 
7 allouer  $S_i$  à l'agent  $i$ 
8 si  $|N| = 2$  alors
9   allouer tous les autres éléments à l'agent restant
10 sinon
11   pour tout  $k \in N \setminus \{i\}$  faire
12     pour tout  $e \in R \setminus S_i$  faire
13        $u_k(e) \leftarrow \frac{u_k(e)}{1 - u_k(S_i)}$  /* normalisation avant la prochaine itération */
14   ALLOCATE ( $R \setminus S_i, N \setminus \{i\}, (u_k)_{k \in N \setminus \{i\}}$ )

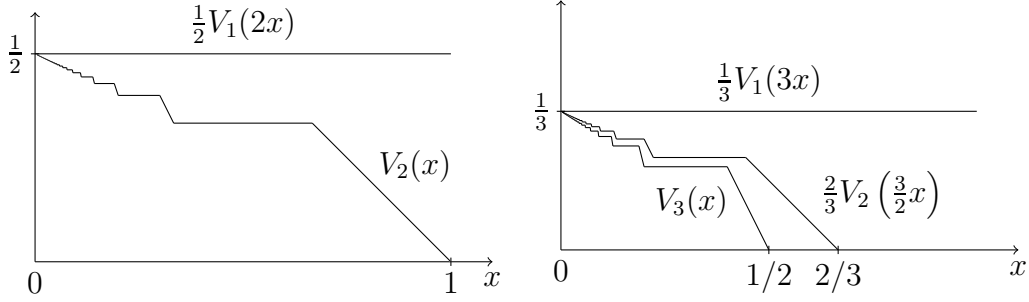
```

---

### 2.3.2 Une propriété clé sur $V_n$

L'objectif de cette section est de donner une estimation (borne inférieure) sur  $\frac{k}{n}V_k\left(\frac{n}{k}x\right)$  pour  $n \geq 2$  et  $n > k \geq 1$  car cette quantité a une importance capitale dans le chapitre 5. La question que nous posons est la suivante : de combien  $\frac{k}{n}V_k\left(\frac{n}{k}x\right)$  est meilleur que  $V_n(x)$  ? Une représentation graphique des fonctions  $\frac{k}{n}V_k\left(\frac{n}{k}x\right)$  est donnée dans la figure 2.2 pour  $n = 2, 3$ .

Pour ce faire, nous introduisons pour tous les entiers  $n, k$  tels que  $n \geq 2$  et


 FIGURE 2.2 – Les fonctions  $\frac{k}{n}V_k\left(\frac{n}{k}x\right)$  et  $V_n(x)$  pour  $n \in \{2, 3\}$  et  $k \in \{1, \dots, n-1\}$ .

$n > k \geq 1$ , une fonction définie par morceaux  $Z_{n,k}(x)$  sur  $[0, 1]$ .

$$Z_{n,k}(x) = \begin{cases} \frac{(n-k)p^2}{n(p+1)((p+1)k-1)}x, & x \in I(n, p) \cup NI(n, p) = \left] \frac{1}{(p+1)^{n-1}}, \frac{1}{pn-1} \right], p \geq 1 \\ 0, & x \in \left] \frac{1}{n-1}, 1 \right] \end{cases}$$

La fonction  $Z_{n,k}$  est croissante en  $p$  pour  $x \in \left[0, \frac{1}{n-1}\right]$ . Puisque  $p \geq 1$ , il s'ensuit que  $Z_{n,k}(x) \geq \frac{n-k}{2n(2k-1)}x$  pour tout  $x \in \left[0, \frac{1}{n-1}\right]$ . De plus,  $\frac{n-k}{2n(2k-1)}x$  est strictement décroissante sur  $k \leq n-1$ , donc  $Z_{n,k}(x) \geq \frac{n-k}{2n(2k-1)}x \geq \frac{x}{2n(2n-3)}$ . Par exemple,  $Z_{2,1}(x) \geq \frac{x}{4}$  et  $Z_{3,k}(x) \geq \frac{x}{18}$  pour tout  $x \in \left[0, \frac{1}{n-1}\right]$  et  $k = 1, 2$ .

**Lemme 2.2** *Étant donnés deux entiers  $n, k$  tels que  $n \geq 2$  et  $n > k \geq 1$ , et un réel  $x \in I(n, p) \cup NI(n, p)$  où  $p \geq 1$  est entier, alors*

- I.  $\frac{k}{n}V_k\left(\frac{n}{k}x\right) \geq V_n(x) + Z_{n,k}(x) \geq V_n(x)$ ,
- II.  $\frac{k}{n}V_k\left(\frac{n}{k}x\right) \geq \frac{k+1}{n}V_{k+1}\left(\frac{n}{k+1}x\right)$ .

La preuve du Lemme 2.2 est donnée dans l'annexe A, page 175.

En particulier, en utilisant  $Z_{n,k}(x) \geq \frac{x}{2n(2n-3)}$ , on déduit que  $V_n(x) + \frac{x}{2n(2n-3)}$  est une borne inférieure de  $\frac{k}{n}V_k\left(\frac{n}{k}x\right)$ .

## 2.4 Algorithmes d'allocation

Une allocation peut être construite au moyen d'un algorithme qui prend souvent en entrée la ressource, les allocations admissibles, les agents et éventuellement leurs utilités.

Les algorithmes centralisés sont les plus répandus dans la littérature où un coordinateur décide de l'allocation finale. La popularité de ces algorithmes s'explique

par leur simplicité d'implantation en machine et de leur efficacité en temps d'exécution, puisque les coûts de communication entre les agents sont limités (Bouveret, 2007). Si le coordinateur est bien choisi (indépendant des  $n$  agents), alors ces algorithmes peuvent être jugés équitables pour les agents, en ne favorisant aucun d'eux. Cependant, ils peuvent également être jugés partiaux lorsque le coordinateur est l'un des  $n$  agents, ou encore dictatoriaux du fait d'imposer une allocation aux agents.

En choix social computationnel, une réflexion autour d'un concept plus élaboré qu'un algorithme centralisé classique est faite. L'objectif est de faire interagir les agents de façon à ce que l'allocation finale soit construite collectivement. On parlera alors d'algorithmes décentralisés (ou distribués), ou encore de *protocoles* de partage. Du point de vue de cette approche, les agents prennent part dans l'élaboration de la solution finale, ce qui les différencie de l'approche MO où un décideur agit pour le bien-être collectif indépendamment de l'intérêt de chaque agent.

### 2.4.1 Algorithmes décentralisés et protocoles de partage

Un protocole est constitué d'un ensemble de règles de communication entre les agents. Les agents se voient faire des propositions de la part des autres agents qu'ils peuvent accepter ou non. Cette approche permet de construire une solution qui est plus facilement acceptée par les agents puisqu'ils contribuent à son choix. Les protocoles décentralisés sont également utiles lorsque les agents ne sont pas capables d'exprimer leur préférences sur des portions de la ressource, ou encore s'ils ne souhaitent pas communiquer certaines informations aux autres agents comme leurs utilités. L'approche décentralisée est également utile lorsqu'une implémentation (au sens informatique) est difficile, voire impossible à réaliser.

Nous mentionnons dans ce qui suit quelques exemples de protocoles de partage de ressources.

#### Ressource divisible

Lorsque la ressource est divisible, il est souvent difficile, voire impossible, de communiquer ses préférences à une entité centrale. La recherche s'oriente alors vers les algorithmes décentralisés et les protocoles de partage de gâteau (problème 19, page 200).

Historiquement, le premier protocole de partage proposé est le paradigme CUT AND CHOOSE pour  $n = 2$  agents. Le premier agent coupe le gâteau en deux parts qu'il juge égales et le second agent choisit la part qu'il préfère, puis le premier agent prend la part restante. Il est évident que ce protocole assure au moins la moitié du gâteau au second agent puisqu'il choisit en premier, et le premier agent aura

la moitié du gâteau selon sa mesure puisqu'il juge les deux parts égales. Ainsi, ce protocole permet d'obtenir une allocation qui est à la fois proportionnelle et sans envie (Brams et Taylor, 1996).

Pour  $n \geq 3$ , plusieurs protocoles proportionnels ont été proposés dès les années 1940 (Steinhaus, 1948). Nous mentionnons par exemple le protocole du *couteau mobile* (*moving-knife*) de Dubins et Spanier (1961). Ce protocole consiste à déplacer lentement un couteau au dessus du gâteau (le traverser sans le toucher) de gauche à droite jusqu'à ce qu'un agent dise stop. Le couteau est alors enfoncé dans le gâteau et la part de gauche reviendra à l'agent qui a dit stop en premier. Puis la procédure est refaite, à chaque fois, avec la partie droite du gâteau et les agents restants, jusqu'à ce qu'il reste un seul agent qui prendra le reste du gâteau. Ce protocole assure au moins  $1/n^{\text{ème}}$  du gâteau à chacun puisque les agents n'ont pas intérêt à dire stop avant que la partie gauche ne vaille au moins  $1/n$  selon leur jugement. De plus, ils n'ont pas intérêt à dépasser le  $1/n^{\text{ème}}$  de leur part car ils risquent d'être lésés si d'autres agents les précèdent en criant stop et que les agents qui ont attendu longtemps se retrouvent avec des parts restantes très petites.

Le premier protocole sans envie pour  $n = 3$  agents (protocole de Selfridge-Conway) a été proposé dans les années 1960. L'agent 1 divise le gâteau en trois parts qu'il juge égales. L'agent 2 passe son tour s'il juge que les deux plus grandes parts sont égales. Dans ce cas, les agents choisissent les parts dans l'ordre 3,2,1. Autrement, l'agent 2 rééquilibre le partage en coupant l'excédant de la plus grande part. Puis les agents choisissent les parts dans l'ordre 3,2,1 sachant que l'agent 2 devra prendre la grande part qu'il a rééquilibré si cette dernière est disponible. Il a fallu trois décennies avant de généraliser ce protocole pour tout  $n \geq 3$  (Brams et Taylor, 1996).

Par la suite, la majorité des travaux sur le *cake-cutting* s'est portée sur l'obtention d'un partage proportionnel, sans envie, etc. en minimisant le nombre d'opérations pour l'atteindre. Brams et Taylor (1996); Robertson et Webb (1998); Procaccia (2013) retracent la majorité des protocoles de partage de gâteaux existants.

### Ressource constituée de biens indivisibles

Dans ce cas, il n'existe pas toujours d'allocation qui satisfasse certains critères d'équité. De même, le problème d'existence de telles allocations peut être NP-complet (comme pour la proportionnalité, l'absence d'envie, l'équitabilité, etc.). Les algorithmes d'approximation (centralisés) apparaissent alors comme une solution potentielle. Nguyen *et al.* (2013) retracent, dans leur état de l'art et dans la thèse de Nguyen (2013), les principaux algorithmes d'approximation connus pour le problème classique d'allocation de biens indivisibles.

De façon générale, il est encore plus difficile de trouver des algorithmes décentralisés pour obtenir une allocation qui satisfait certaines garanties pour les biens indivisibles. Comme alternative, quelques hypothèses restrictives ont été relâchées. Par exemple, le fait de permettre la compensation monétaire entre les agents. Cela permet de compenser la perte d'utilité des agents désavantagés dans une allocation. Dans certains cas, la monnaie est assimilée à une ressource divisible. Ainsi, les agents auront à partager un ensemble de biens indivisibles et la monnaie qui est divisible (Brams et Taylor, 1996). On peut également supposer que chaque agent bénéficie initialement d'une somme d'argent (Estivie, 2006). Puis, selon la procédure d'allocation appliquée, les agents pourront compenser monétairement les agents désavantagés. L'utilité d'un agent sera calculée en fonction des biens qui lui sont alloués ainsi que de la somme d'argent qu'il détient à la fin de la procédure. Il existe plusieurs protocoles qui se basent sur la compensation monétaire comme la procédure des soumissions scellées proposée par Knaster (*Knaster's procedure of sealed bids*) (Steinhaus, 1948) ou encore la procédure de parage (*trimming procedure*) proposée par Brams et Taylor (1996).

La procédure des soumissions scellées de Knaster permet d'obtenir un partage proportionnel en permettant la compensation monétaire. Elle a été proposée dans les années 1940 et repose sur les enchères. Les agents soumettent des propositions scellées sur les biens indivisibles et chaque bien est attribué à l'agent ayant fait la meilleure offre. Puis, une réévaluation de l'utilité de chacun permet de déterminer les agents ayant un excès d'utilité (par rapport au critère proportionnel  $OPT_i/n$ ) qu'ils devront reverser aux agents déficitaires (dont l'utilité est inférieure strictement à  $OPT_i/n$ ) afin que leurs parts soient également proportionnelles.

Nous mentionnons également la procédure de parage (*trimming procedure*) de Brams et Taylor (1996) qui s'inspire du protocole de Selfridge-Conway et qui permet de garantir l'absence d'envie pour l'allocation de biens indivisibles avec compensation monétaire. Dans ce protocole, les agents ont des utilités pour les biens et l'instance est normalisée. Un premier agent partitionne les biens en  $n$  parts auxquelles il peut associer des sommes d'argent afin que son utilité soit la même pour toutes ces parts. Puis, chaque agent (sauf le premier et le dernier), à tour de rôle, rééquilibre le partage de l'agent précédent en modifiant un nombre limité de parts afin de refléter au mieux son sens de l'équité. Par la suite, les agents choisissent les parts en suivant l'ordre inverse que celui établi pour partager les biens, c'est-à-dire en commençant par l'agent qui n'a fait aucun partage des biens, puis celui qui a rééquilibré les parts en dernier et en finissant par l'agent qui a partitionné les parts en premier.

Une autre hypothèse (différente de la compensation monétaire) a été faite dans la méthode des marqueurs de Lucas décrite par Brams et Taylor (1996). Cette procédure s'inspire du protocole du *couteau mobile* de Dubins et Spanier (1961)

pour garantir la proportionnalité dans le cas de biens indivisibles. Ce protocole se base sur une hypothèse forte de linéarité où chaque agent peut partager les biens indivisibles en  $n$  parts égales si les biens sont disposés de façon alignée. La méthode des marqueurs opère comme suit. Commencer par aligner les biens indivisibles, puis chaque agent doit partitionner les éléments en  $n$  parts qu'il juge égales en plaçant  $n - 1$  marqueurs entre les biens alignés sans changer leur ordre. Les marqueurs des agents sont caractérisés différemment. En partant de gauche, on traverse les biens indivisibles un par un jusqu'à atteindre le premier marqueur. Le premier segment à gauche sera alloué à l'agent qui aura placé le premier marqueur rencontré. Puis on supprime les éléments sélectionnés, l'agent qui les a reçus ainsi que ses marqueurs. On recommence l'opération jusqu'à atteindre le deuxième marqueur d'un agent parmi ceux restants. Puis le segment compris entre les deux premiers marqueurs de cet agent lui reviendra. Et ainsi de suite jusqu'à ce qu'il y ait un seul agent qui prendra le segment de droite. Notons qu'il peut rester des éléments non alloués, mais cela n'empêche pas que la solution partielle obtenue soit proportionnelle.

Une autre forme assez répandue de protocoles décentralisés sont les protocoles de négociation. Ces derniers permettent de parvenir à un accord (une allocation) suite à une séquence de négociations et de transactions (échange de ressources) entre les agents (Chevaleyre *et al.*, 2006). Plusieurs transactions sont possibles, selon la possibilité que les négociations soient faites entre deux agents (*one-to-one*), ou qu'un groupe d'agents négocie avec un seul agent (*many-to-one*) qui est le cas typique des enchères, ou encore que les échanges se fassent entre deux groupes d'agents (*many-to-many*). Plusieurs transactions avec des propriétés intéressantes ont été proposées par Sandholm (1998); Andersson et Sandholm (1998).

L'un des protocoles de négociation les plus populaires est l'approche *Contract-Net* (Smith, 1980) qui s'inspire des enchères. Dans ce protocole, un manager déclare une portion de la ressource aux agents. Les agents intéressés par cette portion font une offre. Puis, le manager sélectionnera la meilleure offre et la portion sera affectée en conséquence. Un exemple bien connu est la procédure des soumissions scellées de Knaster décrite précédemment. Il existe plusieurs extensions de ce protocole selon les transactions permises. À titre d'exemple, dans le système TRACONET de Sandholm (1993), les agents font des offres financières afin de négocier des échanges de portions de la ressource avec d'autres agents. Dans le protocole de Golfarelli *et al.* (1997), les agents ne font pas d'offres financières mais proposent des portions de ressource à échanger (Chevaleyre *et al.*, 2006).

Dans ce type d'approches, l'objectif est souvent de montrer qu'après une certaine séquence de négociations, on finit par converger vers des solutions satisfaisant des notions d'équité comme l'utilitarisme (Endriss *et al.*, 2006), l'égalitarisme (Estivie *et al.*, 2005), l'absence d'envie (Chevaleyre *et al.*, 2007c) ou encore l'utilitarisme et l'absence d'envie simultanément (Chevaleyre *et al.*, 2007a) (en per-

mettant la compensation monétaire). Ce type d'approches est développé de façon détaillée dans la thèse d'Estivie (2006).

Nous trouvons également des algorithmes aléatoires comme le protocole de Bezáková et Dani (2005) pour  $n = 2$  agents qui est une version aléatoire de CUT AND CHOOSE pour les biens indivisibles. Ici, un premier agent est choisi en jouant à pile ou face. Puis, le premier agent partage les biens indivisibles en deux parts. La partition est faite de façon à être localement optimale, c'est-à-dire que même si on déplace un élément d'une part vers l'autre, cela ne décroît pas la disparité de la partition (la différence des utilités des deux parts en valeur absolue). Cela peut être réalisé en temps polynomial (Bezáková et Dani, 2005). Puis, le second agent fait l'allocation, c'est-à-dire qu'il choisit en premier sa part préférée, et le premier agent prend la part restante. Ce protocole garantit une espérance d'utilité d'au moins la moitié de l'allocation optimale selon le critère égalitariste. Demko et Hill (1988) proposent également un algorithme pour  $n = 2$  agents qui consiste à allouer  $m - 1$  biens indivisibles de façon déterministe et ils organisent une loterie pour allouer le dernier bien. Cela permet de garantir une espérance d'utilité d'au moins  $\frac{1}{2}$  à chaque agent.

### 2.4.2 Algorithmes à véracité garantie

Dans certains SMA, il est possible que les agents puissent déclarer des utilités qui ne reflètent pas leur préférences réelles afin de s'assurer une meilleure part. Les *mécanismes d'incitation (mechanism design)* traitent ce type de problèmes et le but est souvent d'inciter les agents à déclarer leurs vraies utilités. Il s'agit par exemple de chercher des algorithmes qui sont à *véracité garantie (Strategy proof, dits aussi truthful* dans Lipton *et al.* (2004); Bezáková et Dani (2005); Markakis et Psomas (2011)) où le fait de déclarer les vraies utilités des agents est une stratégie dominante, c'est-à-dire que les agents n'ont aucun intérêt à communiquer de fausses utilités. Ou bien montrer qu'il n'existe pas de procédure à véracité garantie lorsqu'on cherche à satisfaire une ou plusieurs notions d'équité.

Par exemple, Bezáková et Dani (2005) montrent qu'il n'existe pas d'algorithme à véracité garantie qui calcule une allocation optimale selon le critère égalitariste. Lipton *et al.* (2004) montrent qu'il n'existe pas d'algorithme à véracité garantie qui détermine une allocation sans envie s'il en existe ou qui soit d'envie minimum.





Deuxième partie

Contributions



## Équité à un élément près

Ce chapitre traite du problème classique de partage équitable de biens indivisibles et de sa généralisation aux matroïdes. Les notions d'équité considérées sont l'équitabilité, la proportionnalité et l'absence d'envie. Il est bien connu que certaines instances n'admettent pas d'allocations équitables et qu'il est NP-complet de décider si de telles allocations existent. Cependant, une solution proche de l'équité peut exister si une relaxation appropriée de l'équité est adoptée. Dans ce chapitre, nous adaptons les concepts d'équité mentionnés ci-dessus aux matroïdes. Nous proposons également une notion relaxée de l'équité, dite *proche* de l'équité, qui consiste à respecter l'équité à un élément près. Nous montrons qu'une solution proche de l'équité existe toujours et peut être obtenue en temps polynomial dans le contexte général des matroïdes, et par conséquent pour le problème classique d'allocation de biens indivisibles.

Ce chapitre a fait l'objet d'un article paru dans les actes de la conférence internationale «The 21st European Conference on Artificial Intelligence (ECAI'14)» (Gourvès *et al.*, 2014) et une version étendue est en soumission dans une revue internationale.

### Sommaire

---

<b>2.1</b>	<b>Problème <i>générique</i> d'allocation de ressources . . . . .</b>	<b>43</b>
2.1.1	Problème <i>classique</i> d'allocation de ressources . . . . .	48
2.1.2	Généralisation du problème classique aux matroïdes . . . . .	49
<b>2.2</b>	<b>Notions de bien-être social et d'équité . . . . .</b>	<b>54</b>
<b>2.3</b>	<b>Garantie dans le pire cas . . . . .</b>	<b>62</b>
2.3.1	Les fonctions $V_n$ . . . . .	63
2.3.2	Une propriété clé sur $V_n$ . . . . .	65
<b>2.4</b>	<b>Algorithmes d'allocation . . . . .</b>	<b>66</b>
2.4.1	Algorithmes décentralisés et protocoles de partage . . . . .	67
2.4.2	Algorithmes à véracité garantie . . . . .	71

---

## Introduction

Le problème classique d'allocation de biens indivisibles a beaucoup été étudié en informatique (Demko et Hill, 1988; Brams et Taylor, 1996; Lipton *et al.*, 2004; Golovin, 2005; Bouveret et Lang, 2008). Ce problème est défini sur un ensemble  $R$  de  $m$  biens indivisibles et  $n$  agents. Chaque agent  $i \in \{1, \dots, n\}$  a une utilité positive et additive  $u_i(e) \geq 0$  pour chaque bien  $e \in R$ . Notons  $[n] = \{1, \dots, n\}$ . Nous supposons que toutes les instances sont normalisées à 1, c'est-à-dire  $u_i(R) = 1$  pour tout  $i \in [n]$ . L'objectif est de trouver une allocation, dite aussi partition,  $S$  de  $R$  en  $\{S_1, \dots, S_n\}$  où  $S_i$  est la part de l'agent  $i \in [n]$ . Par souci d'équité, nous considérons trois notions : l'équitabilité, la proportionnalité et l'absence d'envie. Une allocation  $\{S_1, \dots, S_n\}$  est équitable si  $u_i(S_i) = u_j(S_j)$  pour tout  $i \neq j$ , elle est proportionnelle si  $u_i(S_i) \geq \frac{1}{n}$  pour tout  $i \in [n]$ , et elle est sans envie si  $u_i(S_i) \geq u_i(S_j)$  pour tout  $i \neq j$ . Ces notions d'équité ont été détaillées dans la section 2.2, page 54.

Comme nous l'avons mentionné précédemment, nous nous intéressons à un problème qui généralise l'allocation de biens indivisibles. Ce problème est défini par un matroïde  $\mathcal{M} = (X, \mathcal{F})$  où une solution réalisable est une base du matroïde. Étant donné  $n \geq 2$  agents avec des utilités positives et additives  $u_i(e) \geq 0$  pour tout  $e \in X$  et pour tout  $i \in [n]$ , l'objectif est de trouver un vecteur de  $n$  ensembles deux à deux disjoints  $(B_1, \dots, B_n)$  où  $B_i$  est la part de l'agent  $i \in [n]$ , et dont l'union  $B = \cup_{i \in [n]} B_i$  est une base de  $\mathcal{M}$ . Ce problème peut modéliser plusieurs situations, certaines d'entre elles sont citées dans la section 2.1.2 (page 49) ou encore dans Lee et Ryan (1992). Par souci d'équité, il serait intéressant d'utiliser les notions d'équité définies pour le problème classique d'allocation de biens indivisibles. Cependant, ces définitions ne sont pas directement applicables au problème générique d'allocation de ressources (problème MA, voir la section 2.1, page 43), et au cas particulier des matroïdes. Une définition appropriée est nécessaire. Dans ce chapitre, nous adoptons les définitions de l'équitabilité, la proportionnalité et l'absence d'envie aux matroïdes.

Il est bien connu qu'il n'existe pas toujours d'allocation du problème classique qui soit équitable, proportionnelle ou sans envie (lorsque tous les biens doivent être alloués). En effet, considérons l'instance simple du problème classique avec deux agents et un seul bien (exemple 2.8, page 57). Allouer ce bien à l'un des deux agents génère une envie et pas d'équitabilité à celui qui n'a pas l'objet, de plus, la proportionnalité ne peut être atteinte puisque le bien est indivisible. Par ailleurs, l'existence d'allocations proportionnelle ou sans envie ne peut être décidé en temps polynomial (voir la section 2.2, page 54). De même, nous montrons dans ce chapitre que décider de l'existence d'une allocation équitable est aussi NP-complet.

En raison de ces constats négatifs, nous nous intéressons à relaxer ces notions d'équité. Nous proposons une notion relaxée dite *proche* de l'équité. Être proche

de l'équité signifie respecter l'équité à un élément près. L'idée d'atteindre l'équité à un élément près apparaît dans Budish (2011) qui introduit *l'envie bornée par un seul élément* dans le contexte du problème classique d'allocation de biens indivisibles sans envie. Une allocation vérifie l'envie bornée par un seul élément si pour toute paire d'agents  $i \neq j$ , soit  $i$  n'envie pas  $j$ , soit il existe un bien dans la part de l'agent  $j$  tel qu'en le supprimant,  $i$  n'est plus envieux de  $j$ . Parkes *et al.* (2012) étudient un problème d'allocation de ressources multiples avec demandes. Ils disent qu'un mécanisme est sans envie à un élément près si pour tout vecteur de valeurs  $b$ , le mécanisme retourne une allocation telle que pour tout  $i, j \in [n]$ ,  $u_i(S_i) \geq u_i(S_j - b_i)$ . Par ailleurs, Lipton *et al.* (2004) s'intéressent à des allocations avec envie bornée et ils présentent un algorithme polynomial qui construit une allocation pour laquelle l'envie est bornée par l'utilité marginale (d'un seul élément)  $\alpha = \max_{S,p,i} u_p(S \cup \{i\}) - u_p(S)$ . Les travaux de Demko et Hill (1988) sur le problème classique évoquent également l'équité à un élément près. Pour deux agents, ils montrent comment allouer tous les biens sauf un de façon déterministe et d'arranger une loterie pour le bien restant afin de garantir une espérance d'utilité de  $\frac{1}{2}$  pour les deux agents. De même, nous proposons certaines définitions afin d'atteindre l'équitabilité, la proportionnalité et l'absence d'envie à un élément près. Ces définitions permettent d'une part, d'assurer l'existence de solutions qui satisfont ces notions d'équité, et d'autre part, de les obtenir en temps polynomial. Nos définitions sont adaptées au problème général défini sur les matroïdes.

La contribution de ce chapitre est d'abord une redéfinition des notions de l'équitabilité, de la proportionnalité et de l'absence d'envie dans le contexte général des matroïdes. Nous introduisons également l'absence de jalousie. De plus, nous proposons des définitions relaxées de ces notions à un élément près, dites proches de l'équité où l'équité signifie un des concepts parmi {absence de jalousie, proportionnalité, absence d'envie}. Nous mettons également en évidence certaines relations entre ces concepts. Nous proposons des algorithmes polynomiaux pour trouver une allocation proche de la jalousie ou proche de la proportionnalité pour des matroïdes. Nous présentons également un algorithme polynomial décentralisé pour obtenir une allocation proche de l'absence d'envie dans le cas particulier de deux agents.

Ce chapitre est organisé comme suit : La section 3.1 présente le modèle. L'équitabilité est le sujet de la section 3.2, l'absence de jalousie est traitée dans la section 3.3, la proportionnalité est considérée dans la section 3.4 et l'absence d'envie dans la section 3.5. Quelques perspectives sont discutées dans la conclusion.

### 3.1 Le modèle

Soit un matroïde  $\mathcal{M} = (X, \mathcal{F})$  où  $X = \{e_1, \dots, e_m\}$ . Sans perte de généralité, nous supposons que pour tout  $e \in X$ ,  $\{e\} \in \mathcal{F}$ . Donc chaque  $e \in X$  appartient à au moins une base (grâce à l'axiome (iii) de la définition 1.12, page 24). Soit un ensemble  $N$  de  $n \geq 2$  agents et une utilité  $u_i(e) \in \mathbb{R}^+$  pour tout  $(i, e) \in N \times X$ . L'utilité d'un agent  $i$  pour un sous-ensemble  $F$  de  $X$  est noté  $u_i(F)$  et est défini par  $\sum_{e \in F} u_i(e)$  (additivité). Par convention  $u_i(\emptyset) = 0$ .

Une solution du problème est appelée *base-allocation*. Une base-allocation est une base  $B$  de  $\mathcal{M}$  partitionnée en  $n$  sous-ensembles  $\{B_1, \dots, B_n\}$  où  $B_i$  est la part de l'agent  $i$ . Nous nous intéressons à l'existence et l'obtention de bases-allocations proches de l'équité  $\in$  {équitabilité, proportionnalité, absence d'envie}.

L'utilité maximum d'un agent  $i \in N$  pour une base de  $\mathcal{M}$  est notée  $OPT_i(\mathcal{M})$ . Nous supposons, pour tout agent  $i$ , que l'instance est normalisée à 1, c'est-à-dire qu'il existe une base  $G^i \in \mathcal{F}$  telle que  $u_i(G^i) = OPT_i(\mathcal{M}) = 1$  et  $u_i(F) \leq 1$  pour tout  $F \in \mathcal{F}$ .

Rappelons que la *contraction* de  $\mathcal{M}$  par  $F \in \mathcal{F}$ , notée  $\mathcal{M}/F$ , est le matroïde  $(X \setminus F, \mathcal{F}')$  où  $\mathcal{F}' = \{E \subseteq X \setminus F : E \cup F \in \mathcal{F}\}$ . Le *complément* de  $F$ , noté  $\mathcal{C}(\mathcal{M}, F)$ , consiste en tous les ensembles  $E \subseteq X$  tels que  $F \cup E$  est une base de  $\mathcal{M}$ . Le *complément maximum* de  $F$  pour une fonction d'utilité  $u$ , noté par  $\mathcal{C}^{\max}(\mathcal{M}, F, u)$ , consiste en tous les ensembles  $E \subseteq X$  tels que  $F \cup E$  est une base de  $\mathcal{M}$  et  $u(E)$  est maximum.

Par abus de notation, nous utilisons souvent  $+$  au lieu de  $\cup$ , et  $-$  au lieu de  $\setminus$ .

### 3.2 Équitabilité

La première notion d'équité que nous traitons est l'équitabilité (Brams et Taylor, 1996; Caragiannis *et al.*, 2012). Dans le cas du problème classique de partage de biens indivisibles, l'équitabilité signifie que tous les agents ont exactement la même utilité pour leurs parts respectives. Nous proposons une généralisation de l'équitabilité aux bases d'un matroïde.

**Définition 3.1 (Équitabilité)** *Une base-allocation  $B = \{B_1, \dots, B_n\}$  est équitable si  $u_i(B_i) = u_j(B_j)$  pour toute paire  $(i, j)$  d'agents.*

Il n'existe pas toujours de base-allocation équitable, même dans le cas particulier du matroïde libre (qui permet de modéliser le problème classique d'allocation de biens indivisibles). De plus, le problème de décision associé est NP-complet.

**Théorème 3.1** *Décider de l'existence d'une base-allocation équitale est un problème  $\mathbb{NP}$ -complet pour un nombre constant d'agents et fortement  $\mathbb{NP}$ -complet dans le cas général, même pour le matroïde libre.*

**Preuve.** Nous supposons que l'oracle d'indépendance qui détermine si un sous-ensemble  $F$  appartient à  $\mathcal{F}$  s'exécute en temps polynomial, autrement le problème ne serait pas dans  $\mathbb{NP}$ .

Nous prouvons la  $\mathbb{NP}$ -complétude pour  $n = 2$  agents en utilisant une réduction à partir de PARTITION et le résultat est valide trivialement pour tout  $n \geq 3$ .

PARTITION : soit un ensemble fini d'éléments  $A = \{a_1, \dots, a_m\}$  et un poids  $s(a) \in \mathbb{Z}^+$  pour tout  $a \in A$ . Existe-t-il un sous-ensemble  $A' \subseteq A$  tel que  $\sum_{a \in A'} s(a) = \sum_{e \in A \setminus A'} s(a)$  ?

Soient l'ensemble  $A = \{a_1, \dots, a_m\}$  et  $(s(a_i))_{i \in [m]} \in \mathbb{Z}^{+m}$  une instance de PARTITION. Définissons  $s(A') = \sum_{e \in A'} s(e)$ ,  $\forall A' \subseteq A$ . Nous construisons une instance du matroïde libre telle que  $A$  est l'ensemble des éléments du matroïde où tous les éléments doivent être alloués. Il y a 2 agents avec des utilités  $u_1(a) = u_2(a) = s(a)$ ,  $\forall a \in A$ . Il est simple de vérifier qu'il existe une base-allocation  $A = \{A_1, A_2\}$  avec  $u_1(A_1) = u_2(A_2)$  si et seulement s'il existe un sous-ensemble  $A' \subseteq A$  avec  $s(A') = s(A \setminus A')$ .

Pour la  $\mathbb{NP}$ -complétude au sens fort du cas général (lorsque le nombre d'agents n'est pas borné), nous faisons une réduction de 3-PARTITION.

3-PARTITION : Soient un ensemble fini  $E$  de  $3k$  éléments, une borne  $b \in \mathbb{Z}^+$  et une fonction poids  $w(e) \in \mathbb{Z}^+$  pour chaque  $e \in E$  telle que  $b/4 < w(e) < b/2$  et  $\sum_{e \in E} w(e) = kb$ . L'ensemble  $E$  peut-il être partitionné en  $k$  ensembles disjoints  $E_1, \dots, E_k$  tels que  $|E_i| = 3$  et  $\sum_{e \in E_i} w(e) = b$  pour tout  $i \in [k]$  ?

Posons  $n = k$  et soit  $\mathcal{M} = (E, \mathcal{F})$  le matroïde libre associé à l'ensemble  $E$ , c'est-à-dire  $\mathcal{F} = 2^E$ . Posons  $u_i(e) = w(e)$ ,  $\forall e \in E$  et  $\forall i \in [n]$ . Nous voulons prouver que  $\{E'_1, \dots, E'_n\}$  est une base-allocation de  $\mathcal{M}$  avec  $u_i(E'_i) = u_i(E'_j)$ ,  $\forall i, j \in [n]$  si et seulement si  $E$  peut être partitionné en  $E_1, \dots, E_k$  tel que  $|E_i| = 3$  et  $\sum_{e \in E_i} w(e) = b$  pour tout  $i \in [k]$ .

Supposons que  $E$  peut être partitionné en  $E_1, \dots, E_k$  tels que  $|E_i| = 3$  et  $\sum_{e \in E_i} w(e) = b$  pour tout  $i \in [k]$ . Alors le résultat est trivial en posant  $E'_i = E_i$ ,  $\forall i \in [n]$ .

Réciproquement, supposons qu'il existe une base-allocation  $E = \{E'_1, \dots, E'_n\}$  telle que  $u_i(E'_i) = u_i(E'_j)$ ,  $\forall i, j \in [n]$ . Posons  $E_i = E'_i$  pour tout  $i \in [k]$ . D'une part,  $w(E_i) = u_i(E'_i) = b$  pour tout  $i \in [k]$  puisque  $nu_i(E_i) = \sum_{e \in E} w(e) = nb$ . D'autre part, il est simple de montrer que  $|E_i| = |E_j| = 3$  pour tout  $i \in [k]$  car  $b/4 < u_i(e) = w(e) < b/2$  pour tout  $e \in E$  et pour tout  $i \in [k]$ . ■

Il existe des cas particuliers où le problème peut être résolu en temps polynomial.



**Proposition 3.1** *On peut décider en temps polynomial si une instance admet une base-allocation équitable lorsque*

1.  $u_i(B_i) = 0$  pour tout  $i \in N$  ;
2.  $u_i(B_i) = 1$  pour tout  $i \in N$ .

**Preuve.**

1. Le problème peut être résolu par le problème de la base de poids minimum (voir page 31). Posons pour tout  $e \in X$ ,  $u(e) = \min_{i \in N} u_i(e)$ , puis appliquer GLOUTON  $(\mathcal{M}, u, \min)$  (voir l'algorithme 1.1, page 32) afin d'obtenir une base  $B$  telle que  $u(B)$  est minimum. La réponse à l'existence d'une base-allocation équitable est oui si et seulement si  $u(B) = 0$ . En effet, si  $u(B) = 0$  alors la partition est simple à réaliser, il suffit d'allouer chaque élément  $e \in B$  à n'importe quel agent  $i$  vérifiant  $u_i(e) = 0$  afin d'obtenir  $u_i(B_i) = 0$  pour tout  $i \in N$ . Si  $u(B) > 0$  alors il n'existe pas de base-allocation  $B' = \{B'_1, \dots, B'_n\}$  telle que  $u_i(B'_i) = 0$  pour tout  $i \in N$  car autrement le GLOUTON (algorithme 1.1 page 32) aurait retourné une base dont le poids est le même que celui de  $B'$  puisque  $u(B') = 0 < u(B)$ .
2. Nous pouvons résoudre ce cas par le problème de la base de poids maximum. Posons pour tout  $e \in X$ ,  $u(e) = \max_{i \in N} u_i(e)$ , puis appliquer GLOUTON  $(\mathcal{M}, u, \max)$  afin d'obtenir une base  $B$  telle que  $u(B)$  est maximum. Il existe une base-allocation équitable si et seulement si  $u(B) = n$ . Si  $u(B) = n$  alors la partition est obtenue en prenant  $B_i = \{e \in B : u(e) = u_i(e)\}$  pour tout  $i \in N$  et tel que  $B_i \cap B_j = \emptyset$  pour tout  $i \neq j$ . Puisque  $u_i(B_i) \leq 1$  (car l'instance est normalisée à 1) et  $\sum_{i \in N} u_i(B_i) = u(B) = n$ , on conclut que  $u_i(B_i) = 1$  pour tout  $i \in N$ . Si  $u(B) < n$  alors il n'existe pas de base-allocation  $B' = \{B'_1, \dots, B'_n\}$  telle que  $u_i(B'_i) = n$  pour tout  $i \in N$  puisqu'autrement, le GLOUTON aurait retourné une base dont le poids est le même que celui de  $B'$  car  $u(B') = n > u(B)$ . ■

Maintenant, nous supposons qu'il existe deux valeurs possibles pour les utilités, c'est-à-dire  $u_i(e) \in \{a, b\}$  avec  $a = 0 < b$  pour tout  $i \in N$  et pour tout  $e \in X$ .

**Proposition 3.2** *Si pour un  $v > 0$ , l'utilité d'un agent pour un élément est soit 0, soit  $v > 0$ , alors on peut décider en temps polynomial si une base-allocation équitable existe.*

**Preuve.** L'instance est  $(\mathcal{M}, N, (u_i)_{i \in N})$  et le rang de  $\mathcal{M} = (X, \mathcal{F})$  est noté  $r$ . Supposons que  $u_j(x) \in \{0, v\}$  pour tout  $(j, x) \in N \times X$ . Dans une base-allocation équitable, l'utilité de chaque agent est égale à  $kv$  où  $k$  est un entier vérifiant  $0 \leq k \leq \lfloor \frac{r}{n} \rfloor$ .

Considérons le matroïde-allocation  $(Y, \mathcal{E})$  défini sur  $\mathcal{M} = (X, \mathcal{F})$  (voir page 51). Pour tout  $j \in N$ , soit  $V_j^v = \{y_i^j \in Y : u_j(x_i) = v\}$ . Soit  $V^0 = Y \setminus \left(\bigcup_{j \in N} V_j^v\right)$ . Les ensembles  $V^0, V_1^v, V_2^v, \dots, V_n^v$  forment une partition de  $Y$ . Pour un certain  $k \in \left[0, \lfloor \frac{r}{n} \rfloor\right]$ , considérons le matroïde de partition  $(Y, \mathcal{H}_k)$  où  $\mathcal{H}_k = \{S \subseteq Y : |S \cap V^0| \leq r - nk \text{ et } |S \cap V_j^v| \leq k, \forall j \in N\}$ .

Nous résolvons ce problème en utilisant l'intersection des deux matroïdes  $(Y, \mathcal{E})$  et  $(Y, \mathcal{H}_k)$  de cardinalité maximum (voir page 33). En effet, observons qu'un ensemble indépendant  $I \in \mathcal{E} \cap \mathcal{H}_k$  de cardinalité  $r$  existe si et seulement si  $(X, \mathcal{F})$  admet une solution équitable  $p(I)$  où l'utilité de chaque agent est  $kv$ .

Puisque  $(Y, \mathcal{E})$  et  $(Y, \mathcal{H}_k)$  sont deux matroïdes définis sur le même ensemble  $Y$ , nous pouvons trouver, en temps polynomial, un ensemble indépendant  $I \in \mathcal{E} \cap \mathcal{H}_k$  de cardinalité maximum. Si  $|I| = r$  alors  $p(I)$  est une base de  $(X, \mathcal{F})$  telle que  $u_j(p(I)) = kv$  pour tout  $j \in N$ . Ainsi,  $I$  est équitable.

La procédure décrite ci-dessus est réalisée pour un  $k$  fixé. De façon générale, nous pouvons procéder de sorte que pour toute valeur entière de  $k \in \left[0, \lfloor \frac{r}{n} \rfloor\right]$ , tester si une solution équitable existe (avec la procédure ci-dessus) et si c'est le cas, on l'obtient en temps polynomial. ■

La complexité du cas où les utilités des agents pour les éléments sont soit  $a$  ou  $b$  avec  $0 < a < b$  est un problème ouvert.

### 3.3 Absence de jalousie

Nous nous intéressons dans cette section à la notion d'absence de jalousie, introduite dans la section 2.2, page 59. La jalousie exprime le mécontentement d'un agent lorsqu'un tiers a une plus grande utilité que la sienne. En effet, un agent  $i$  est jaloux d'un agent  $j \neq i$  si l'utilité de l'agent  $i$  est strictement inférieure à celle de l'agent  $j$ . Nous introduisons la notion d'absence de jalousie afin de relaxer l'équitabilité.

**Définition 3.2 (Absence de jalousie)** *Soit la base-allocation  $B = \{B_1, \dots, B_n\}$ .*

- *L'agent  $i$  est sans jalousie par rapport à l'agent  $j$  lorsque  $u_i(B_i) \geq u_j(B_j)$  ;*
- *L'agent  $i$  est sans jalousie lorsque  $u_i(B_i) \geq u_j(B_j)$  pour tout  $j \in N$  ;*
- *Une base-allocation est sans jalousie lorsque tous les agents sont sans jalousie.*

Une base-allocation sans jalousie est équivalente à une base-allocation équitable.

Nous introduisons une relaxation de l'absence de jalousie à un élément près, dite *proche de l'absence de jalousie*.

**Définition 3.3 (Proche de l'absence de jalousie)** Soit la base-allocation  $B = \{B_1, \dots, B_n\}$ .

- L'agent  $i$  est proche de l'absence de jalousie par rapport à l'agent  $j$  lorsque  $u_i(B_i) \geq u_j(B_j - e)$  pour tout  $e \in B_j$  ;
- L'agent  $i$  est proche de l'absence de jalousie lorsqu'il l'est par rapport à tous les agents  $j \in N$  ;
- Une base-allocation est proche de l'absence de jalousie si tous les agents sont proches de l'absence de jalousie.

Un agent  $i$  est proche de l'absence de jalousie lorsqu'il est sans jalousie par rapport à tous les agents, ou bien si  $i$  est jaloux d'un agent  $j$ , alors en supprimant n'importe quel élément  $e$  de  $B_j$ , l'agent  $i$  n'est plus jaloux de  $j$ .

**Observation 3.1** L'agent  $i$  est proche de l'absence de jalousie par rapport à l'agent  $j$  si et seulement si

$$u_i(B_i) \geq u_j(B_j) - \min_{e \in B_j} u_j(e)$$

**Théorème 3.2** Une base-allocation proche de l'absence de jalousie existe toujours et peut être obtenue en  $O(n|X| \max\{\ln |X|, \theta\})$ .

**Preuve.** Montrons que la solution retournée par l'algorithme 3.1 est proche de l'absence de jalousie. Dans l'algorithme 3.1, la solution est construite en choisissant l'agent (étape 3) ou l'élément (étape 4) avec un indice minimum.

---

**Algorithme 3.1 : PROCHE DE L'ABSENCE DE JALOUSIE**

---

**Données :** un matroïde  $\mathcal{M} = (X, \mathcal{F})$ ,  $n = |N|$  agents et des utilités  $(u_i)_{i \in N}$

**Résultat :** une base-allocation  $B$  proche de l'absence de jalousie

- 1  $B \leftarrow \emptyset$  et  $B_i \leftarrow \emptyset$  pour tout  $i \in N$
  - 2 **tant que**  $B$  n'est pas une base de  $\mathcal{M}$  **faire**
  - 3     trouver un  $i \in N$  tel que  $u_i(B_i) = \min_{j \in N} u_j(B_j)$
  - 4     trouver  $e \in X \setminus B$  tel que  $B + e \in \mathcal{F}$  et  $u_i(e)$  est maximum
  - 5      $B \leftarrow B + e$  et  $B_i \leftarrow B_i + e$
  - 6 **retourner**  $B = \{B_1, \dots, B_n\}$
- 

L'algorithme 3.1 trouve une base de  $\mathcal{M}$  et son temps d'exécution est dominé par les  $n$  tris des éléments de  $X$  et les tests d'indépendance réalisés en  $O(|X| \max\{\ln |X|, \theta\})$  (étape 4).

Soient  $i, j \in N$  deux agents et sans perte de généralité, supposons que  $u_i(B_i) \geq u_j(B_j)$ . Considérons le dernier élément  $e \in X$  ajouté à  $B_i$ . Cet élément correspond à une itération où les parts des agents  $i$  et  $j$  étaient respectivement  $B'_i = B_i - e$  et  $B'_j \subseteq B_j$ . Par construction, l'agent  $i$  a été sélectionné afin de recevoir un nouvel élément  $e$ , donc  $u_i(B'_i) \leq u_j(B'_j)$ . Chaque nouvel élément inséré (étape 4) est le

plus grand parmi l'ensemble des éléments qui peuvent être ajoutés et qui satisfont la propriété d'indépendance (c'est-à-dire  $B'_i + e \in \mathcal{F}$ ). Ainsi,  $u_i(e) = \min_{y \in B_i} u_i(y)$ . Par ailleurs,  $u_i(B_i) - \min_{y \in B_i} u_i(y) = u_i(B'_i) \leq u_j(B'_j) \leq u_j(B_j)$ , ce qui signifie que les agents  $i$  et  $j$  sont tous les deux proches de l'absence de jalousie. ■

L'algorithme 3.1 est inspiré de l'algorithme de Graham (1969) pour minimiser le temps total d'exécution (*makespan*) de  $m$  tâches sur  $n$  machines identiques et parallèles. L'algorithme de Graham (1969) a également été utilisé par Coffman et Langston (1984); Lipton *et al.* (2004) pour montrer une 1.4-approximation pour minimiser le ratio d'envie dans le problème classique, lorsque les agents ont tous la même fonction d'utilité.

**Exemple 3.1** Soit le matroïde de partition  $\mathcal{M} = (X, \mathcal{F})$  (voir page 30) défini par  $X = \cup_{i=1}^3 X_i$  avec  $X_1 = \{e_1\}$ ,  $X_2 = \{e_2\}$ ,  $X_3 = \{e_3, e_4, e_5\}$  et  $\mathcal{F} = \{F \subseteq X : |F \cap X_1| \leq 1, |F \cap X_2| \leq 1, |F \cap X_3| \leq 2\}$ . Il y a 3 agents, leurs utilités sont données dans le tableau 3.1.

	$X_1$	$X_2$	$X_3$		
$e$	$e_1$	$e_2$	$e_3$	$e_4$	$e_5$
$u_1(e)$	0.4	0.25	0.2	0.15	0.15
$u_2(e)$	0.3	0.2	0.2	0.2	0.3
$u_3(e)$	0.25	0.25	0.25	0.25	0.25

Tableau 3.1 – Exemple 3.1.

Il n'existe aucune base-allocation sans jalousie. Cependant, l'algorithme 3.1 fournit la base-allocation  $B_1 = \{e_1\}$ ,  $B_2 = \{e_5\}$  et  $B_3 = \{e_2, e_3\}$  qui est proche de l'absence de jalousie. En effet,

$$\begin{aligned}
 u_1(B_1) &= 0.4 > u_2(B_2) - \min_{e \in B_2} u_2(e) = 0, \\
 u_1(B_1) &= 0.4 > u_3(B_3) - \min_{e \in B_3} u_3(e) = 0.25, \\
 u_2(B_2) &= 0.3 > u_1(B_1) - \min_{e \in B_1} u_1(e) = 0, \\
 u_2(B_2) &= 0.3 > u_3(B_3) - \min_{e \in B_3} u_3(e) = 0.25, \\
 u_3(B_3) &= 0.5 > u_1(B_1) - \min_{e \in B_1} u_1(e) = 0, \\
 u_3(B_3) &= 0.5 > u_2(B_2) - \min_{e \in B_2} u_2(e) = 0.
 \end{aligned}$$

## 3.4 Proportionnalité

Nous nous intéressons dans cette section à la notion de proportionnalité. Cette dernière signifie que chaque agent reçoit au moins  $1/n^{\text{ème}}$  de la ressource totale

qu'il est possible de lui allouer (voir la section 2.2, page 56). Nous proposons dans cette section une définition de la proportionnalité adaptée aux bases d'un matroïde. Rappelons que nous supposons que  $OPT_i = 1$ ,  $i = 1, \dots, n$ .

**Définition 3.4 (Proportionnalité)**

- Une base-allocation  $B = \{B_1, \dots, B_n\}$  est proportionnelle pour l'agent  $i$  lorsque  $u_i(B_i) \geq \frac{1}{n}$ ;
- Une base-allocation est proportionnelle lorsque elle est proportionnelle pour tous les agents.

Il n'existe pas toujours de base-allocation proportionnelle. De plus, décider de l'existence d'une base-allocation proportionnelle pour un matroïde donné est NP-complet puisqu'elle généralise l'existence d'une allocation proportionnelle du problème classique de partage de biens indivisibles qui est NP-complet (Demko et Hill, 1988; Bouveret et Lemaître, 2014). Nous proposons une relaxation de la proportionnalité à un élément près, dite *proche de la proportionnalité*.

**Définition 3.5 (Proche de la proportionnalité)**

- Une base-allocation  $B = \{B_1, \dots, B_n\}$  est proche de la proportionnalité pour un agent  $i$  s'il existe  $f \in B$  et  $e \in X$  tels que  $(B - f) + e$  est une base et  $u_i((B_i - f) + e) \geq \frac{1}{n}$ ;
- Une base-allocation est proche de la proportionnalité lorsqu'elle est proche de la proportionnalité pour tous les agents.

Une base-allocation est proche de la proportionnalité pour un agent  $i$  si elle est proportionnelle ou si l'agent  $i$  peut interchanger un élément de la base-allocation et obtenir une nouvelle part qui est proportionnelle. Cela signifie qu'il existe un élément  $f \in B$  tel qu'en le supprimant, l'agent  $i$  peut ajouter un nouvel élément  $e$  dans sa part et la base-allocation devient proportionnelle. Notons que l'élément  $f$  n'est pas forcément dans  $B_i$ . Dans ce cas,  $(B_i - f) + e = B_i + e$ .

Une base-allocation proche de la proportionnalité existe toujours et peut être obtenue avec l'algorithme 3.2 qui utilise l'algorithme 3.3.

L'algorithme 3.3 s'inspire de l'algorithme ALLOCATE (algorithme 2.1, page 65) de Markakis et Psomas (2011) pour le problème classique d'allocation de biens indivisibles avec une garantie dans le pire cas sur l'utilité de chaque agent (voir la section 2.3, page 62). L'algorithme 3.3 alloue aux agents de  $N_1$  des parts proches de la proportionnalité. Cet algorithme est récursif et à chaque étape de la récursion, on choisit un agent qui atteint la proportionnalité de l'instance restante avec le moins d'éléments possible. Cet agent se voit allouer la part qu'il aura sélectionnée. Puis, tant qu'il reste des agents dans  $N_1$ , on relance l'algorithme avec les agents restants, leurs utilités et le matroïde contracté de la part de l'agent sélectionné précédemment.

**Algorithme 3.2 : PROCHE DE LA PROPORTIONNALITÉ****Données** : un matroïde  $\mathcal{M} = (X, \mathcal{F})$ ,  $n = |N|$  agents et des utilités  $(u_i)_{i \in N}$ **Résultat** : une base-allocation proche de la proportionnalité  $B$ 

- 1  $N_1 \leftarrow \{i \in N : \max_{e \in X} u_i(e) \leq \frac{1}{n}\}$  et  $N_2 \leftarrow N - N_1$
- 2  $F_1 \leftarrow \emptyset$  et  $F_2 \leftarrow \emptyset$
- 3 **si**  $N_1 \neq \emptyset$  **alors**
- 4    $F_1 \leftarrow$  Algorithme 3.3 ( $\mathcal{M}, n, N_1, (u_i)_{i \in N_1}$ )
- 5 **si**  $N_2 \neq \emptyset$  **alors**
- 6   soit  $F_2$  une base de  $\mathcal{M}/F_1$
- 7   allouer aléatoirement les éléments de  $F_2$  aux agents de  $N_2$
- 8 **retourner**  $B = F_1 \cup F_2$

**Algorithme 3.3 : PROCHE DE LA PROPORTIONNALITÉ POUR LES AGENTS DE  $N_1$** **Données** : un matroïde  $\hat{\mathcal{M}}$ , un entier  $\hat{n}$ , un ensemble d'agents  $\hat{N}$  et des utilités  $(u_i)_{i \in \hat{N}}$ 

- 1 **pour tout**  $i \in \hat{N}$  **faire**
- 2   soit  $G^i \leftarrow$  GROUTON ( $\mathcal{M}, u_i, \max$ ) avec  $G^i = \{e_1^i, \dots, e_{r(\hat{\mathcal{M}})}^i\}$  et  
 $u_i(e_1^i) \geq \dots \geq u_i(e_{r(\hat{\mathcal{M}})}^i)$
- 3    $OPT_i(\hat{\mathcal{M}}) \leftarrow u_i(G^i)$
- 4   soit  $k_i$  le plus grand indice tel que  $u_i(\{e_1^i, \dots, e_{k_i}^i\}) \leq \frac{OPT_i(\hat{\mathcal{M}})}{\hat{n}}$
- 5 choisir un agent  $\ell \in \hat{N}$  tel que  $k_\ell \leq k_i$ , pour tout  $i \in \hat{N}$
- 6  $B_\ell \leftarrow \{e_1^\ell, \dots, e_{k_\ell}^\ell\}$
- 7 **si**  $|\hat{N}| = 1$  **alors**
- 8   **retourner**  $B_\ell$
- 9 **sinon**
- 10   **retourner**  $B_\ell \cup$  Algorithme 3.3 ( $\hat{\mathcal{M}}/B_\ell, \hat{n} - 1, \hat{N} - \ell, (u_j)_{j \in \hat{N} - \ell}$ )

Il est possible d'utiliser le GROUTON (algorithme 1.1, page 32) à l'étape 6 de l'algorithme 3.2. La façon dont les éléments de  $F_2$  sont alloués à l'étape 7 de l'algorithme 3.2 n'a pas d'incidence sur l'analyse de la base-allocation résultante.

**Théorème 3.3** *Une base-allocation proche de la proportionnalité existe toujours et peut être obtenue en  $O(n^2|X| \max\{\ln |X|, \theta\})$ .*

**Preuve.** Analysons  $B$ , la solution retournée par l'algorithme 3.2. Posons  $n = |N|$  et  $n_i = |N_i|$  pour  $i = 1, 2$ . Donc  $n = n_1 + n_2$ .

Commençons par  $N_2$ . Lorsqu'on arrive à l'étape 5 de l'algorithme 3.2, cela signifie que  $N_1 = \emptyset$ . Dans ce cas, si  $N_2 \neq \emptyset$  alors par construction, il existe un élément dont l'utilité est au moins  $1/n$  pour chaque agent de  $N_2$ . Considérons un agent  $j \in N_2$  et soit  $B_j$  sa part. Soit  $e \in X$  tel que  $u_j(e) \geq 1/n$  et soit  $B'$

une base qui contient  $e$  (chaque élément appartient à une base). Si  $e \in B$ , alors  $(B - e) + e$  est une base et  $u_j((B_j - e) + e) \geq u_j(e) \geq 1/n$ . Sinon,  $e \in B' - B$  et en utilisant le théorème 1.3 (page 27), il existe  $f \in B$  tel que  $(B - f) + e$  est une base. De plus,  $u_j((B_j - f) + e) \geq u_j(e) \geq 1/n$ . Par conséquent,  $B$  est proche de la proportionnalité pour chaque agent  $j \in N_2$ .

Considérons maintenant les agents de  $N_1$  et supposons que  $n_1 \geq 1$ . Leurs parts sont construites avec l'algorithme 3.3 qui est récursif. L'algorithme 3.3 est exécuté  $|N_1|$  fois et à chaque fois, un agent de  $N_1$  reçoit sa part (voir l'étape 6).

Afin de montrer que  $B$  est proche de la proportionnalité pour chaque membre de  $N_1$ , nous suivons l'ordre selon lequel les agents se voient allouer leurs parts respectives. Par souci de simplicité, et sans perte de généralité, renommons ces agents  $1, \dots, n_1$ . Cela signifie que l'agent  $i \in N_1$  se voit allouer  $B_i$  durant le  $i^{\text{ème}}$  appel récursif de l'algorithme 3.3.

Montrons d'abord que pour tout  $i, i' \in N_1$  vérifiant  $i \leq i'$ , on a

$$OPT_{i'}(\mathcal{M}/(\bigcup_{j < i} B_j)) \geq \frac{n - i + 1}{n} \quad (3.1)$$

La preuve est par récurrence sur  $i$ . Lorsque  $i = 1$  alors  $OPT_{i'}(\mathcal{M}) \geq 1$  pour tout  $i' \in N_1$  grâce à l'hypothèse de normalisation. Supposons que l'inégalité (3.1) est vraie pour  $i = 1..p$ . Considérons le  $p^{\text{ème}}$  appel récursif de l'algorithme 3.3 dans lequel l'agent  $p$  se voit finalement allouer  $B_p$ . Dans ce cas,  $\hat{\mathcal{M}} = \mathcal{M}/(\bigcup_{j < p} B_j)$ ,  $\hat{n} = n - p + 1$  et  $\hat{N} = \{p, \dots, n_1\}$ . En raison de l'étape 5,  $k_{i'} \geq k_p$  pour tout  $i' > p$ ;  $u_{i'}(G^{i'}) = OPT_{i'}(\hat{\mathcal{M}})$  en utilisant l'étape 3 et  $u_{i'}(\{e_1^{i'}, \dots, e_{k_{i'}}^{i'}\}) \leq \frac{OPT_{i'}(\hat{\mathcal{M}})}{\hat{n}}$  grâce à l'étape 4. Il s'ensuit que

$$u_{i'}(G^{i'}) - u_{i'}(\{e_1^{i'}, \dots, e_{k_{i'}}^{i'}\}) \geq \frac{\hat{n} - 1}{\hat{n}} OPT_{i'}(\hat{\mathcal{M}}) \quad (3.2)$$

En utilisant le lemme 1.2 (page 33) avec  $F_2 = G^{i'}$ ,  $F_1 = B_p$ , et  $u = u_{i'}$ , on obtient  $OPT_{i'}(\hat{\mathcal{M}}/B_p) \geq u_{i'}(G^{i'} \setminus \{e_1^{i'}, \dots, e_{k_{i'}}^{i'}\})$ . En utilisant l'additivité de  $u_{i'}$ , et le fait que  $G^{i'} \supseteq \{e_1^{i'}, \dots, e_{k_{i'}}^{i'}\}$ , on déduit que

$$OPT_{i'}(\hat{\mathcal{M}}/B_p) \geq u_{i'}(G^{i'}) - u_{i'}(\{e_1^{i'}, \dots, e_{k_{i'}}^{i'}\})$$

En utilisant l'inégalité (3.2) et sachant que  $\hat{n} = n - p + 1$ , on obtient

$$OPT_{i'}(\hat{\mathcal{M}}/B_p) \geq \frac{\hat{n} - 1}{\hat{n}} OPT_{i'}(\hat{\mathcal{M}}) = \frac{n - p}{n - p + 1} OPT_{i'}(\hat{\mathcal{M}})$$

En utilisant l'hypothèse de récurrence,

$$OPT_{i'}(\hat{\mathcal{M}}) = OPT_{i'}(\mathcal{M}/\bigcup_{j < p} B_j) \geq \frac{n - p + 1}{n}$$

Il s'ensuit que

$$OPT_{i'}(\mathcal{M} / \bigcup_{j < p+1} B_j) = OPT_{i'}(\hat{\mathcal{M}}/B_p) \geq \frac{n-p}{n-p+1} \frac{n-p+1}{n} = \frac{n-p}{n}$$

La propriété est valide pour  $i = p + 1$  et l'inégalité (3.1) est prouvée.

Il reste à montrer que  $B$  est proche de la proportionnalité pour tout agent  $i \in N_1$ . Durant le  $i^{\text{ème}}$  appel de l'algorithme 3.3, l'agent  $i$  reçoit finalement  $B_i$ . De plus,  $\hat{\mathcal{M}} = \mathcal{M}/(\bigcup_{j < i} B_j)$ ,  $\hat{n} = n - i + 1$  et  $\hat{N} = \{i, \dots, n_1\}$ . À l'étape 2 de l'algorithme 3.3,  $G^i$  est une base optimale de  $\hat{\mathcal{M}}$  pour  $u_i$ , donc  $u_i(G^i) = OPT_i(\hat{\mathcal{M}})$ . En utilisant l'inégalité (3.1), on obtient  $u_i(G^i) \geq \frac{n-i+1}{n}$ . Par construction,  $B_i$  consiste en les  $k_i$  éléments de poids maximum de  $G^i$  pour  $u_i$ , qui sont  $\{e_1^i, \dots, e_{k_i}^i\}$ . Si  $k_i = |G^i|$  alors  $u_i(B_i) = u_i(G^i) \geq \frac{n-i+1}{n} \geq \frac{1}{n}$  (la dernière inégalité est due à  $i \leq n$ ); donc  $B$  est proche de la proportionnalité pour l'agent  $i$ . Supposons que  $k_i < |G^i|$ . On obtient  $u_i(\{e_1^i, \dots, e_{k_i}^i\}) \leq \frac{OPT_i(\hat{\mathcal{M}})}{\hat{n}}$  grâce à l'étape 4 et il existe  $e_{k_i+1}^i$  tel que  $u_i(\{e_1^i, \dots, e_{k_i+1}^i\}) > OPT_i(\hat{\mathcal{M}})/\hat{n}$ . Puisque  $OPT_i(\hat{\mathcal{M}}) = u_i(G^i) \geq \frac{n-i+1}{n}$  et  $\hat{n} = n - i + 1$ , on obtient  $u_i(\{e_1^i, \dots, e_{k_i+1}^i\}) > 1/n$ .

Si  $e_{k_i+1}^i \in B$ , alors  $(B - e_{k_i+1}^i) + e_{k_i+1}^i$  est une base de  $\mathcal{M}$  et  $u_i((B_i - e_{k_i+1}^i) + e_{k_i+1}^i) \geq u_i(\{e_1^i, \dots, e_{k_i+1}^i\}) \geq 1/n$ . Sinon,  $e_{k_i+1}^i \notin B$  et en utilisant le théorème 1.3 (page 27), il existe  $f \in B$  tel que  $(B - f) + e_{k_i+1}^i$  est une base de  $\mathcal{M}$ . De plus  $u_i((B_i - f) + e_{k_i+1}^i) \geq u_i(\{e_1^i, \dots, e_{k_i+1}^i\}) \geq 1/n$ , donc  $B$  est proche de la proportionnalité pour l'agent  $i$ .

Pour conclure, la complexité de l'algorithme 3.2 est due aux  $O(n^2)$  appels du GLOUTON (algorithme 1.1, page 32) de complexité  $O(|X| \max\{\ln |X|, \theta\})$ . ■

**Exemple 3.2** L'instance présentée dans l'exemple 3.1 (page 83) n'admet pas de base-allocation proportionnelle. Cependant, une base-allocation proche de la proportionnalité peut être obtenue avec l'algorithme 3.2. À l'étape 1,  $N_1 = \{2, 3\}$  et  $N_2 = \{1\}$ . Puis appliquer l'algorithme 3.3 sur  $\mathcal{M}$ ,  $n = 3$ ,  $N_1$  et  $(u_i)_{i \in N_1}$ .

Dans la première itération de l'algorithme 3.3,  $G^2 = \{e_1, e_5, e_2, e_3\}$  et  $G^3 = \{e_1, e_2, e_3, e_4\}$ , donc  $k_2 = k_3 = 1$ ,  $\ell = 2$  et  $B_2 = \{e_1\}$ .

La seconde itération de l'algorithme 3.3 est sur  $\hat{\mathcal{M}} = \mathcal{M}/\{e_1\} = (X - e_1, \mathcal{F}')$  avec  $\mathcal{F}' = \{F \subseteq X - e_1 : |F \cap X_2| \leq 1, |F \cap X_3| \leq 2\}$ ,  $\hat{n} = 2$ ,  $\hat{N}_1 = \{3\}$  et  $u_3$ . On obtient  $G^3 = \{e_2, e_3, e_4\}$ ,  $k_3 = 1$ ,  $\ell = 3$  et  $B_3 = \{e_2\}$  tel que  $u_3(B_3) = 0.25 < \frac{OPT_3(\hat{\mathcal{M}})}{\hat{n}} = \frac{u_3(G^3)}{2} = \frac{0.75}{2}$ . Finalement,  $F_1 = B_2 \cup B_3$ .

Par exemple, l'ensemble  $F_2 = \{e_3, e_4\}$  est une base de  $\mathcal{M}/F_1$  qui peut être obtenue avec le GLOUTON (algorithme 1.1, page 32). Il reste à allouer les éléments de  $F_2$  à l'unique agent de  $N_2$  qui est l'agent 1, donc  $B_1 = F_2$ . Finalement, la base résultante est  $B = F_1 \cup F_2 = \{B_1, B_2, B_3\}$  et  $B$  est proche de la proportionnalité. En effet,



$$\begin{aligned} u_1(B_1) &= u_1(\{e_3, e_4\}) = 0.35 > \frac{1}{3}, \\ u_2(B_2) &= u_2(\{e_1\}) = 0.3 < \frac{1}{3} \text{ mais } u_2((B_2 - e_2) + e_2) = 0.5 > \frac{1}{3}, \\ u_3(B_3) &= u_3(\{e_2\}) = 0.25 < \frac{1}{3} \text{ mais } u_3((B_3 - e_1) + e_1) = 0.5 > \frac{1}{3}. \end{aligned}$$

### 3.5 Absence d'envie

Dans ce qui suit, nous généralisons la définition de l'absence d'envie aux matroïdes (voir la section 2.2, page 57).

**Définition 3.6 (Absence d'envie)** *Soit une base-allocation  $B = \{B_1, \dots, B_n\}$ .*

- *L'agent  $i$  est sans envie par rapport à l'agent  $j$  lorsque  $u_i(B_i) \geq u_i(B'_j)$  pour tout  $B'_j \in \mathcal{C}(\mathcal{M}, B - B_j)$  où  $\mathcal{C}(\mathcal{M}, B - B_j)$  est le complément de  $B - B_j$ , c'est-à-dire qu'il contient tous les ensembles  $B'_j \subseteq X$  tels que  $B - B_j \cup B'_j$  est une base de  $\mathcal{M}$  ;*
- *L'agent  $i$  est sans envie lorsqu'il n'envie aucun agent  $j \in N$  ;*
- *Une base-allocation est sans envie lorsque tous les agents sont sans envie.*

Si l'agent  $i$  a l'opportunité de compléter  $B - B_j$  en une base, comme l'agent  $j$  l'a fait pour sa part  $B_j$ , alors n'importe quel sous-ensemble  $B'_j$  ne serait pas meilleur que  $B_i$ , du point de vue de l'agent  $i$ . Dans le cas particulier de l'allocation de biens indivisibles,  $B_j$  correspond aux biens qui ne sont pas alloués aux agents de  $N \setminus \{j\}$ , et qui est l'unique façon de compléter  $B - B_j$ . Par ailleurs, la définition 3.6 dans ce contexte établit que l'agent  $i$  ne serait pas envieux de la part de l'agent  $j$ , qui est cohérent avec la notion standard d'absence d'envie. En effet, nous aurions pu définir l'absence d'envie par  $u_i(B_i) \geq u_i(B_j)$  pour chaque paire  $(i, j)$  d'agents, cependant la définition 3.6 est plus forte.

**Observation 3.2** *Si une base-allocation  $B = \{B_1, \dots, B_n\}$  est sans envie, alors pour tout  $i \in N$ ,*

$$u_i(B_i) = OPT_i(\mathcal{M}/(B - B_i)) \tag{3.3}$$

**Preuve.** Une base-allocation  $B = \{B_1, \dots, B_n\}$  est sans envie si et seulement si  $u_i(B_i) \geq u_i(B'_j)$  pour tout  $B'_j \in \mathcal{C}(\mathcal{M}, B - B_j)$  et pour tout  $j \in N$ . Ceci est vrai en particulier pour  $j = i$ , donc  $u_i(B_i) \geq OPT_i(\mathcal{M}/(B - B_i))$ . ■

Nous montrons, dans ce qui suit, que l'absence d'envie implique la proportionnalité et que l'équivalence est valable pour deux agents pour les matroïdes, comme pour le problème classique d'allocation de biens divisibles / indivisibles avec des utilités additives.

**Proposition 3.3** *L'absence d'envie implique la proportionnalité lorsque  $n \geq 2$ . De plus, si  $n = 2$ , alors la proportionnalité implique l'absence d'envie.*

**Preuve.** Supposons que la base-allocation  $B = \{B_1, \dots, B_n\}$  est sans envie, c'est-à-dire pour tout  $i, j \in N$ ,  $u_i(B_i) \geq u_i(B'_j)$  pour tout  $B'_j \in \mathcal{C}(\mathcal{M}, B \setminus B_j)$ . Ceci est vrai en particulier pour  $B'_j = G_j^i$  où  $G^i = G_1^i \cup \dots \cup G_n^i$  est une base maximum pour  $u_i$  vérifiant pour tout  $j \in N$ ,  $B - B_j + G_j^i$  est une base (une telle partition de  $G^i$  existe grâce au théorème 1.5, page 27). En sommant sur  $j$ ,  $nu_i(B_i) = \sum_{j=1}^n u_i(B_i) \geq \sum_{j=1}^n u_i(G_j^i) = u_i(G^i) = 1$  car  $G^i$  est maximum pour  $u_i$ . Ainsi,  $u_i(B_i) \geq \frac{1}{n}$  pour tout  $i \in N$ , donc  $B = \{B_1, \dots, B_n\}$  est proportionnelle.

Pour  $n = 2$ , supposons que  $B = \{B_1, B_2\}$  est proportionnelle, c'est-à-dire  $u_i(B_i) \geq \frac{1}{2}$  pour  $i = 1, 2$ . Soit  $B'_j \in \mathcal{C}(\mathcal{M}, B_i)$  pour chaque  $i \in \{1, 2\}$ . Ainsi,  $u_i(B_i \cup B'_j) = u_i(B_i) + u_i(B'_j)$  par additivité et  $B_i \cap B'_j = \emptyset$ , donc  $u_i(B_i \cup B'_j) \leq 1$  puisque l'instance est normalisée à 1. Comme  $u_i(B_i) \geq \frac{1}{2}$ , il s'ensuit que  $u_i(B'_j) \leq \frac{1}{2}$ . Ainsi,  $u_i(B_i) \geq u_i(B'_j)$  pour tout  $B'_j \in \mathcal{C}(\mathcal{M}, B_i)$ , pour  $i = 1, 2$  et  $\{B_1, B_2\}$  est sans envie. ■

Une base-allocation sans envie n'existe pas toujours et décider de l'existence d'une base-allocation sans envie est NP-complet même dans le cas particulier du problème classique d'allocation de biens indivisibles (Lipton *et al.*, 2004). Nous présentons dans ce qui suit une relaxation de l'absence d'envie à un élément près.

**Définition 3.7 (Proche de l'absence d'envie)** *Soit une base-allocation  $B = \{B_1, \dots, B_n\}$ .*

- *L'agent  $i$  est proche de l'absence d'envie par rapport à l'agent  $j$  lorsque  $u_i(B_i) \geq u_i(B'_j)$  pour tout  $B'_j \in \mathcal{C}(\mathcal{M}, (B - B_j) + e)$  où  $e \in X \setminus (B - B_j)$  et  $(B - B_j) + e \in \mathcal{F}$  ;*
- *L'agent  $i$  est proche de l'absence d'envie lorsqu'il est proche de l'absence d'envie de tout agent  $j \in N$  ;*
- *Une base-allocation est proche de l'absence d'envie lorsque chaque agent est proche de l'absence d'envie.*

La définition 3.7 est plus forte que celle de *l'envie bornée par un seul élément* (Budish, 2011) dans le contexte du problème classique d'allocation des biens indivisibles. En effet, si l'agent  $i$  est envieux de l'agent  $j$  alors être proche de l'absence d'envie signifie qu'en supprimant n'importe quel élément de la part de  $j$ , l'agent  $i$  est sans envie par rapport à l'agent  $j$ . L'envie bornée par un seul élément signifie qu'il existe un élément dans la part de l'agent  $j$  tel qu'en le supprimant, l'agent  $i$  élimine son envie pour l'agent  $j$ .

**Proposition 3.4** *L'agent  $i$  est proche de l'absence d'envie par rapport à l'agent  $j$  si et seulement si*

$$u_i(B_i) \geq u_i(D) - \min_{x \in D} u_i(x), \quad \forall D \in \mathcal{C}(\mathcal{M}, B - B_j)$$

**Preuve.** Supposons que  $B = \{B_1, \dots, B_n\}$  vérifie la définition 3.7 et soit  $D \in \mathcal{C}(\mathcal{M}, B - B_j)$ . Supposons que  $e = \arg \min_{x \in D} u_i(x)$ . Alors,  $(B - B_j) + e \in \mathcal{F}$  et  $D - e \in \mathcal{C}(\mathcal{M}, (B - B_j) + e)$ . De la définition 3.7, on déduit que  $u_i(B_i) \geq u_i(D - e) = u_i(D) - u_i(e) = u_i(D) - \min_{x \in D} u_i(x)$ .

Réciproquement, soit  $e \in X \setminus (B - B_j)$  avec  $(B - B_j) + e \in \mathcal{F}$  et soit  $B'_j \in \mathcal{C}(\mathcal{M}, (B - B_j) + e)$ . Posons  $D = B'_j + e$ ; alors  $u_i(e) \geq \min_{x \in D} u_i(x)$ . Ainsi,  $u_i(B_i) \geq u_i(D) - \min_{x \in D} u_i(x) \geq u_i(D) - u_i(e) = u_i(B'_j)$ . ■

**Exemple 3.3** L'instance de l'exemple 3.1 (page 83) n'admet pas de base-allocation sans envie. Cependant, la base-allocation  $B = \{B_1, B_2, B_3\}$  où  $B_1 = \{e_1\}$ ,  $B_2 = \{e_5\}$  et  $B_3 = \{e_2, e_3\}$  est proche de l'absence d'envie. En effet :

1. L'agent 1 est proche de l'absence d'envie pour les raisons suivantes :
  - $C(\mathcal{M}, B - B_1) = \{B_1\}$  donc l'agent 1 vérifie l'égalité (3.3).
  - $C(\mathcal{M}, B - B_2) = \{\{e_4\}, \{e_5\}\}$  et  $u_1(B_1) = 0.4 > u_1(D) - \min_{e \in D} u_1(e) = 0$  pour tout  $D \in C(\mathcal{M}, B - B_2)$ .
  - $C(\mathcal{M}, B - B_3) = \{\{e_2, e_3\}, \{e_2, e_4\}\}$ . Ainsi,  $u_1(B_1) = 0.4 > u_1(\{e_2, e_3\}) - \min_{e \in \{e_2, e_3\}} u_1(e) = 0.25$  et  $u_1(B_1) = 0.4 > u_1(\{e_2, e_4\}) - \min_{e \in \{e_2, e_4\}} u_1(e) = 0.25$ .
2. L'agent 2 est proche de l'absence d'envie pour les raisons suivantes :
  - $C(\mathcal{M}, B - B_1) = \{B_1\}$  et  $u_2(B_2) = 0.3 > u_2(B_1) - \min_{e \in B_1} u_2(e) = 0$ .
  - $C(\mathcal{M}, B - B_2) = \{\{e_4\}, \{e_5\}\}$  et puisque  $B_2 = \{e_5\}$  et que  $u_2(e_5) > u_2(e_4)$  alors l'agent 2 vérifie l'égalité (3.3).
  - $C(\mathcal{M}, B - B_3) = \{\{e_2, e_3\}, \{e_2, e_4\}\}$  et  $u_2(B_2) = 0.3 > u_2(D) - \min_{e \in D} u_2(e) = 0.2$  pour tout  $D \in C(\mathcal{M}, B - B_3)$ .
3. L'agent 3 vérifie l'égalité (3.3) car  $C(\mathcal{M}, B - B_3) = \{\{e_2, e_3\}, \{e_2, e_4\}\}$  et  $u_3(B_3) = u_3(\{e_2, e_3\}) = u_3(\{e_2, e_4\}) = 0.5$ . De plus,  $u_3(B_3) = 0.5 > u_i(B_i)$ ,  $i = 1, 2$ . Ainsi, l'agent 3 est sans envie.

Étant donnée une base-allocation proche de l'absence d'envie, on peut construire en temps polynomial une base-allocation proche de l'absence d'envie vérifiant les égalités (3.3), page 88. En effet, soit  $B = \{B_1, \dots, B_n\}$  une base-allocation proche de l'absence d'envie. Alors pour  $i = 1$  à  $n$ , appliquer  $B'_i := \text{GLOUTON}(\mathcal{M}/((\cup_{j < i} B'_j) \cup (\cup_{k > i} B_k)), u_i, \max)$ . Il est clair que la base-allocation résultante  $B' = \{B'_1, \dots, B'_n\}$  est proche de l'absence d'envie et vérifie les égalités (3.3). La complexité est bornée par les  $n$  exécutions du GLOUTON (algorithme 1.1, page 32).

**Lemme 3.1** Une base-allocation proche de l'absence d'envie et vérifiant les égalités (3.3) est aussi proche de la proportionnalité.

**Preuve.** Soit  $B = \{B_1, \dots, B_n\}$  une base-allocation proche de l'absence d'envie, c'est-à-dire

$$\forall i, j \in N, u_i(B_i) \geq u_i(D) - \min_{x \in D} u_i(x), \quad \forall D \in \mathcal{C}(\mathcal{M}, B - B_j) \quad (3.4)$$

Cela est vrai en particulier pour  $D = G_j^i$  où  $G^i = \cup_{j \in N} G_j^i$  est une base maximum pour  $u_i$  partitionnée en  $n$  parts  $\{G_j^i, \dots, G_j^i\}$  de sorte que  $B - B_k + G_k^i$  est une base pour tout  $k \in N$ . Une telle partition existe grâce au théorème 1.5 (page 27). Soit  $i \in N$ . En sommant les inégalités (3.4) sur  $j \in N \setminus \{i\}$ , on obtient

$$(n-1)u_i(B_i) = \sum_{j \in N \setminus \{i\}} u_i(B_i) \geq \sum_{j \in N \setminus \{i\}} u_i(G_j^i) - \sum_{j \in N \setminus \{i\}} \min_{x \in G_j^i} u_i(x) \quad (3.5)$$

En utilisant l'égalité (3.3),

$$u_i(B_i) = OPT_i(\mathcal{M}/(B - B_i)) \geq u_i(G_i^i) \quad (3.6)$$

En sommant les inégalités (3.5) et (3.6), on obtient

$$nu_i(B_i) \geq \sum_{j \in N} u_i(G_j^i) - \sum_{j \in N \setminus \{i\}} \min_{x \in G_j^i} u_i(x) = 1 - \sum_{j \in N \setminus \{i\}} \min_{x \in G_j^i} u_i(x) \quad (3.7)$$

car  $G^i = \cup_{j \in N} G_j^i$  est maximum pour  $u_i$  et est normalisée à 1.

Posons  $e = \arg \max_{j \in N \setminus \{i\}} \min_{x \in G_j^i} u_i(x)$ . Soit  $k \in N \setminus \{i\}$  tel que  $e \in G_k^i$ . Alors

$$\sum_{j \in N \setminus \{i\}} \min_{x \in G_j^i} u_i(x) \leq (n-1) \max_{j \in N \setminus \{i\}} \min_{x \in G_j^i} u_i(x) = (n-1)u_i(e)$$

L'inégalité (3.7) devient

$$u_i(B_i) \geq \frac{1}{n} - \frac{n-1}{n}u_i(e) \geq \frac{1}{n} - u_i(e) \quad (3.8)$$

Nous savons que  $B_k$  et  $G_k^i$  sont des bases de  $\mathcal{M}/(B - B_k)$ . Si  $e \in B_k$  alors  $B_i + e \subseteq B \in \mathcal{F}$ . En utilisant l'inégalité (3.8) et l'additivité de  $u_i$ , on obtient  $u_i(B_i + e) \geq \frac{1}{n}$  pour tout  $i \in N$ , donc  $B$  est proche de la proportionnalité.

Maintenant, supposons que  $e \notin B_k$ . Appliquons le théorème 1.3 (page 27) sur  $B_k$  et  $G_k^i$  dans le matroïde  $\mathcal{M}/(B - B_k)$  sachant que  $e \in G_k^i \setminus B_k$ . Alors il existe  $f \in B_k$  tel que  $B_k - f + e$  est une base de  $\mathcal{M}/(B - B_k)$ , donc  $B - f + e$  est une base de  $\mathcal{M}$ . Puisque  $f \notin B_i$  (car  $f \in B_k, k \neq i$ ) et en utilisant (3.8) et l'additivité de  $u_i$ , il s'ensuit que  $u_i(B_i - f + e) = u_i(B_i + e) \geq \frac{1}{n}$  pour tout  $i \in N$ , donc  $B$  est proche de la proportionnalité. ■

Notons qu'il existe des instances montrant qu'une base-allocation proche de l'absence d'envie et ne respectant pas les égalités (3.3) n'implique pas qu'elle soit proche de la proportionnalité. L'exemple 3.4 en atteste.

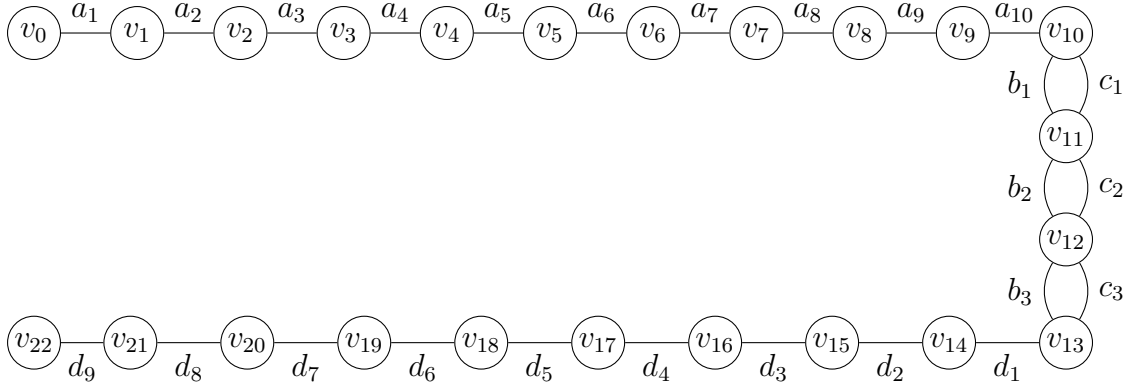


FIGURE 3.1 – Exemple 3.4.

**Exemple 3.4** Considérons le matroïde graphique défini sur le multi-graphe représenté dans la figure 3.1.

Il y a trois agents  $N = \{1, 2, 3\}$  qui ont tous la même fonction d'utilité définie par :  $u(a_i) = u(d_j) = \frac{1}{31}$ ,  $i = 1, \dots, 10$ ,  $j = 1, \dots, 9$ ,  $u(b_k) = \frac{3}{31}$  et  $u(c_k) = \frac{4}{31}$ ,  $k = 1, 2, 3$ . Une base de ce matroïde est un arbre couvrant du graphe. L'arbre couvrant qui contient toutes les arêtes de type  $a_i$ ,  $c_j$  et  $d_k$  constitue une base optimale dont l'utilité est 1.

La base-allocation  $B_1 = \{a_i, i = 1, \dots, 10\}$ ,  $B_2 = \{b_i, i = 1, 2, 3\}$  et  $B_3 = \{d_i, i = 1, \dots, 9\}$  est proche de l'absence d'envie. En effet,  $u(B_1) \geq u(B_i)$ ,  $i = 2, 3$ , c'est-à-dire que l'agent 1 est sans envie,  $u(B_i) \geq u(B_j) - \min_{e \in B_j} u(e)$ ,  $i = 2, 3$ ,  $j \in N - i$ , c'est-à-dire que les agents 2 et 3 sont proches de l'absence d'envie.

L'agent 2 ne vérifie pas l'égalité (3.3) puisque  $u(B_2) = \frac{9}{31}$ , or le meilleur moyen de compléter la forêt  $\{a_i, d_j, i = 1, \dots, 10, j = 1, \dots, 9\}$  est d'ajouter les arêtes  $c_1$ ,  $c_2$  et  $c_3$  dont l'utilité est  $u(\{c_1, c_2, c_3\}) = \frac{12}{31} > u(B_2)$ .

Nous constatons que cette base-allocation n'est pas proche de la proportionnalité par rapport à l'agent 2. En effet, en supprimant une arête quelconque  $b_i$  de  $B_2$ , on ne peut la remplacer que par  $c_i$ , donc  $u(B_2 - b_i + c_i) = \frac{10}{31} < \frac{1}{3}$ . Par ailleurs, en supprimant une arête de type  $a_i$  ou  $b_j$  de  $B$ , cette dernière ne peut être remplacée que par elle-même pour obtenir un arbre couvrant, donc  $u(B_2 - a_i + a_i) = u(B_2) + u(a_i) = \frac{10}{31} < \frac{1}{3}$  car  $a_i \notin B_2$  pour  $i = 1, \dots, 10$ . De même,  $u(B_2 - d_j + d_j) = \frac{10}{31} < \frac{1}{3}$  pour  $j = 1, \dots, 9$ .

Être proche de la proportionnalité n'implique pas être proche de l'absence d'envie même pour deux agents comme dans l'exemple suivant.

**Exemple 3.5** Soit le matroïde libre  $\mathcal{M} = (X, \mathcal{F})$  défini par  $X = \{e_1, e_2, e_3\}$  et  $\mathcal{F} = \{F \subseteq X\}$ . Les utilités sont :  $u_1(e_1) = 0.4$ ,  $u_1(e_2) = u_1(e_3) = 0.3$  et

$u_2(e_i) = \frac{1}{3}$  pour tout  $i \in \{1, 2, 3\}$ . Considérons la base-allocation  $B = \{B_1, B_2\}$  avec  $B_1 = \{e_3\}$  et  $B_2 = \{e_1, e_2\}$ .  $B$  est proche de la proportionnalité puisque  $u_1((B_1 - e_2) + e_2) = 0.6 > \frac{1}{2}$  et  $u_2(B_2) = \frac{2}{3} > \frac{1}{2}$  mais elle n'est pas proche de l'absence d'envie car  $u_1(B_1) = 0.3 < u_1(B_2) - \min_{e \in B_2} u_1(e) = 0.4$ .

### 3.5.1 Proche de l'équilibre

Nous introduisons le concept de *proche de l'équilibre* pour une fonction d'utilité  $u$ . Ce concept est inspiré de l'*optimalité locale* dans les travaux de Bezáková et Dani (2005) dans le contexte du problème classique d'allocation de biens indivisibles (voir la section 2.4.1, page 67).

**Définition 3.8 (Proche de l'équilibre)** *Étant donné un matroïde  $\mathcal{M} = (X, \mathcal{F})$ , une fonction d'utilité  $u : X \rightarrow \mathbb{R}^+$  et un entier  $n \geq 2$ , alors une base-allocation  $B = \{B_1, \dots, B_n\}$  avec  $u(B_1) \geq \dots \geq u(B_n)$  est proche de l'équilibre pour  $u$  si*

$$i < j \Rightarrow u(B_j) \geq u(B_i) - \min_{y \in B_i} u(y), \quad \forall i, j \in [1..n] \quad (3.9)$$

**Corollaire 3.1** *Une base-allocation proche de l'équilibre existe toujours et peut être calculée en temps polynomial.*

**Preuve.** Être proche de l'équilibre correspond à être proche de l'absence de jalousie dans le cas particulier où les agents ont la même fonction d'utilité. Le résultat provient du théorème 3.2 et de l'algorithme 3.1 (page 82). ■

### 3.5.2 Proche de l'absence d'envie pour deux agents

L'algorithme 3.4 présente un algorithme décentralisé qui construit une base-allocation proche de l'absence d'envie pour deux agents.

---

#### Algorithme 3.4 : PROCHE DE L'ABSENCE D'ENVIE POUR DEUX AGENTS

---

**Données :** un matroïde  $\mathcal{M} = (X, \mathcal{F})$ , agents 1 et 2,  $u_1, u_2$

**Résultat :** une base-allocation proche de l'absence d'envie

- 1 L'agent 1 calcule une base  $G^1$  maximum pour  $u_1$  (avec le GLOUTON) et il la partitionne en  $G_1^1$  et  $G_2^1$  de façon à être proche de l'équilibre.
  - 2 L'agent 2 calcule  $V^i \in C^{\max}(\mathcal{M}, G_i^1, u_2)$  pour  $i = 1, 2$  (avec le GLOUTON), puis il choisit  $V^k = \arg \max_{i=1,2} u_2(V^i)$  et l'agent 1 prend  $G_k^1$ .
- 

L'algorithme 3.4 s'inspire du protocole CUT AND CHOOSE pour couper un gâteau en deux (voir la section 2.1.1, page 48).

**Théorème 3.4** *Lorsqu'il y a deux agents, une base-allocation proche de l'absence d'envie existe toujours et peut être obtenue en temps polynomial.*

**Preuve.** Soit  $N = \{1, 2\}$ . Soient  $G^1$  et  $G^2$  deux bases qui maximisent respectivement  $u_1$  et  $u_2$ . Supposons que  $G^1$  est partitionnée en  $G_1^1 \cup G_2^1$  de façon à ce que  $G_1^1$  et  $G_2^1$  soient proches de l'équilibre pour  $u_1$ .

Soient  $V^1 \in \mathcal{C}^{max}(\mathcal{M}, G_1^1, u_2)$  et  $V^2 \in \mathcal{C}^{max}(\mathcal{M}, G_2^1, u_2)$ . Alors  $G_1^1 \cup V^1$  et  $G_2^1 \cup V^2$  sont deux bases de  $\mathcal{M}$ . De plus,  $V^1$  et  $V^2$  sont des bases respectivement de  $\mathcal{M}/G_1^1$  et  $\mathcal{M}/G_2^1$ . Notons que  $V^1 \cup V^2$  n'est pas forcément un ensemble indépendant. Supposons sans perte de généralité que  $u_2(V^1) \geq u_2(V^2)$ . Par construction,  $u_2(V^1) \geq u_2(D)$  pour tout  $D \in \mathcal{C}(\mathcal{M}, G_1^1)$ . Dans ce qui suit, nous analysons la solution  $B = \{G_1^1, V^1\}$ .

Les agents satisfont clairement les égalités (3.3), donc nous avons besoin uniquement de prouver que l'agent  $i \in \{1, 2\}$  est proche de l'absence d'envie par rapport à l'agent  $3 - i$ .

Montrons que l'agent 2 est sans envie par rapport à l'agent 1. En d'autres termes,  $u_2(V^1) \geq u_2(D)$  pour tout  $D \in \mathcal{C}(\mathcal{M}, B \setminus G_1^1) = \mathcal{C}(\mathcal{M}, V^1)$ . Soit  $D \in \mathcal{C}(\mathcal{M}, V^1)$ ; donc  $D \cup V^1$  est une base.

Soit  $G^2$  une base maximum pour  $u_2$ . En utilisant le théorème 1.5 (page 27) avec  $G^1 = G_1^1 \cup G_2^1$  et  $G^2$ , on déduit que  $G^2$  admet une bi-partition  $G_1^2 \cup G_2^2$  telle que  $G_1^2 \cup G_2^1$  et  $G_1^1 \cup G_2^2$  sont deux bases de  $\mathcal{M}$ . Ainsi  $G_1^2$  et  $G_2^2$  appartiennent respectivement à  $\mathcal{C}(\mathcal{M}, G_2^1)$  et  $\mathcal{C}(\mathcal{M}, G_1^1)$ . Puisque  $V^1$  et  $V^2$  appartiennent respectivement à  $\mathcal{C}^{max}(\mathcal{M}, G_1^1, u_2)$  et  $\mathcal{C}^{max}(\mathcal{M}, G_2^1, u_2)$ , alors  $u_2(V^1) \geq u_2(G_2^2)$  et  $u_2(V^2) \geq u_2(G_1^2)$ . En sommant ces deux inégalités,  $u_2(V^1) + u_2(V^2) \geq u_2(G_2^2) + u_2(G_1^2) = u_2(G^2) \geq u_2(D \cup V^1)$ . Ainsi,  $u_2(V^2) \geq u_2(D)$ . Puisque  $u_2(V^1) \geq u_2(V^2)$  par hypothèse, le résultat s'ensuit.

Montrons que l'agent 1 est proche de l'absence d'envie par rapport à l'agent 2.  $G^1$  est partitionnée en  $G_1^1 \cup G_2^1$  telle que  $G_1^1$  et  $G_2^1$  sont proches de l'équilibre pour  $u_1$ . Nous montrons que l'agent 1 est proche de l'absence d'envie en utilisant la proposition 3.4 (page 89). Soit  $D \in \mathcal{C}(\mathcal{M}, G_1^1)$ . Nous savons que  $G_1^1 \cup D$  est une base, donc  $u_1(G_1^1) + u_1(G_2^1) \geq u_1(G_1^1 \cup D)$ . D'où,

$$u_1(G_2^1) \geq u_1(D) \tag{3.10}$$

Nous distinguons deux cas : soit  $u_1(G_1^1) \geq u_1(G_2^1)$  ou  $u_1(G_1^1) < u_1(G_2^1)$ . Évidemment, si  $u_1(G_1^1) \geq u_1(G_2^1)$ , alors l'agent 1 est sans envie car en utilisant l'inégalité (3.10), on obtient  $u_1(G_1^1) \geq u_1(G_2^1) \geq u_1(D)$ .

Nous étudions le second cas, c'est-à-dire  $u_1(G_2^1) > u_1(G_1^1)$ . Puisque  $\{G_1^1, G_2^1\}$  est une base proche de l'équilibre, on a  $u_1(G_1^1) \geq u_1(G_2^1) - \min_{x \in G_2^1} u_1(x)$ . Montrons que  $u_1(G_2^1) - \min_{x \in G_2^1} u_1(x) \geq u_1(D) - \min_{x \in D} u_1(x)$ , ce qui conclurait la preuve.

Soit  $e = \arg \min_{x \in G_2^1} u_1(x)$  et  $f = \arg \min_{x \in D} u_1(x)$  et par contradiction, supposons que  $u_1(D - f) > u_1(G_2^1 - e)$ . En utilisant l'axiome (iii) d'un matroïde (voir la définition 1.12 page 24), il existe  $y \in G^1 - ((D + G_1^1) - f) = G_2^1 - (D - f)$  tel que  $B' = (D - f) + G_1^1 + y$  est une base. Forcément,  $u_1(y) \geq u_1(e)$  car  $y \in G_2^1$  et  $e$  est par définition minimum selon  $u_1$  dans  $G_2^1$ . Ainsi,  $u_1(B') = u_1(D - f) + u_1(G_1^1) + u_1(y) > u_1(G_2^1 - e) + u_1(G_1^1) + u_1(y) = u_1(G^1) - u_1(e) + u_1(y) \geq u_1(G^1)$ , contradiction avec le fait que  $G^1$  est une base maximum pour  $u_1$ . ■

## Conclusion et perspectives

Nous nous sommes intéressés au problème qui consiste à trouver une base d'un matroïde et de la partager équitablement sur un ensemble d'agents. Ce problème généralise le problème classique d'allocation de biens indivisibles. Nous avons introduit la notion d'équité à un élément près, dite proche de l'équité, lorsque l'équité correspond à la proportionnalité, l'équitabilité (ou l'absence de jalousie) et l'absence d'envie.

Une solution équitable peut ne pas exister. Cependant, une solution déterministe proche de l'équité est garantie d'exister et peut être obtenue en temps polynomial. Pour l'équitabilité et la proportionnalité, le résultat est obtenu avec des algorithmes centralisés valides pour n'importe quel nombre d'agents. Est-il possible d'obtenir les mêmes résultats avec des algorithmes décentralisés? Concernant l'absence d'envie, le résultat est obtenu pour deux agents, avec un algorithme décentralisé où les agents ne communiquent pas leurs utilités respectives. Il reste à étudier l'existence d'une base-allocation proche de l'absence d'envie pour  $n \geq 3$  agents. Nous conjecturons qu'une telle base-allocation existe toujours pour tout  $n \geq 3$ . Une autre perspective est de combiner l'équité à un élément près avec d'autres critères standards comme la Pareto-optimalité.





# Garantie dans le pire cas : une approche centralisée

Nous considérons le problème classique d'allocation de biens indivisibles avec des utilités additives. L'objectif est de construire une allocation avec une garantie dans le pire cas sur l'utilité des agents. Demko et Hill (1988) montrent l'existence d'une allocation où l'utilité de chaque agent vaut au moins  $V_n(\alpha)$ , qui est une famille de fonctions, et  $\alpha$  est l'utilité maximum d'un bien. Markakis et Psomas (2011) proposent un algorithme polynomial qui retourne une telle allocation. La garantie  $V_n(\alpha)$  est maximum pour certaines valeurs de  $\alpha$ , c'est-à-dire que c'est la meilleure borne inférieure sur l'utilité de l'agent le moins heureux. Mais ceci n'est pas vrai pour toutes les valeurs de  $\alpha$ . Nous proposons une famille de fonctions  $W_n$  telles que  $W_n(x) \geq V_n(x)$  pour tout  $x$ , et  $W_n(x) > V_n(x)$  pour les valeurs de  $x$  où  $V_n(x)$  n'est pas maximum. Les fonctions  $W_n$  s'appliquent au problème de la base d'un matroïde multi-agents traité dans le chapitre 3 et qui généralise le problème classique d'allocation de biens indivisibles. Nos résultats sont obtenus en analysant une extension de l'algorithme de Markakis et Psomas (2011).

Ce chapitre a été soumis dans une revue internationale. Une version préliminaire apparaît dans les actes de la conférence «The 23rd International Joint Conference on Artificial Intelligence (IJCAI'13)» (Gourvès *et al.*, 2013a).

## Sommaire

---

<b>Introduction</b> . . . . .	<b>76</b>
<b>3.1 Le modèle</b> . . . . .	<b>78</b>
<b>3.2 Équitabilité</b> . . . . .	<b>78</b>
<b>3.3 Absence de jalousie</b> . . . . .	<b>81</b>
<b>3.4 Proportionnalité</b> . . . . .	<b>83</b>
<b>3.5 Absence d'envie</b> . . . . .	<b>88</b>
3.5.1 Proche de l'équilibre . . . . .	93
3.5.2 Proche de l'absence d'envie pour deux agents . . . . .	93
<b>Conclusion et perspectives</b> . . . . .	<b>95</b>

---

## Introduction

Demko et Hill (1988) s'intéressent au problème classique d'allocation de biens indivisibles à  $n$  agents ( $n \geq 2$ ). Les agents peuvent avoir différentes utilités pour les biens indivisibles, et l'utilité d'un agent pour une portion de biens est additive. Après une normalisation à 1, il est supposé que l'utilité de chaque agent pour l'intégralité des biens est 1.

Demko et Hill (1988) cherchent à savoir comment l'agent le moins heureux peut valoriser sa part. Il s'agit de trouver un seuil  $t_n \in [0, 1]$  tel que, dans tous les cas, il existe une allocation où chacun des  $n$  agents a une utilité d'au moins  $t_n$  pour sa part. Le seuil  $t_n$  doit également être accompagné d'une famille d'instances sans allocation réalisable où la part de chacun vaudrait au moins  $t_n + \epsilon$ ,  $\forall \epsilon > 0$ .

Dans le contexte du partage de biens *divisibles*, il est bien connu que  $t_n = 1/n$  (Steinhaus, 1948). Le fait de considérer des biens indivisibles mène à une situation plus délicate. En effet, considérons l'exemple de  $n \geq 2$  agents qui ont une utilité de 1 pour le même bien, noté  $e_1$ , et une utilité de 0 pour tous les autres biens (voir l'exemple 2.8, page 57). Les agents qui ne reçoivent pas  $e_1$  ont une utilité globale de 0, signifiant que  $t_n$  est égal à 0. Mais si l'utilité maximum d'un bien est bornée par une quantité qui tend vers 0, alors le modèle indivisible tendrait progressivement vers le modèle divisible où  $t_n = 1/n$ . Mais quelle est la valeur de  $t_n$  entre ces deux cas extrêmes ?

L'utilité maximum d'un seul bien semble influencer de manière significative  $t_n$ , comme dans l'article pionnier de Hill (1987) qui définit une famille de fonctions décroissantes  $V_n : [0, 1] \rightarrow [0, \frac{1}{n}]$  pour tout  $n \geq 2$  (voir la section 2.3, page 62). En suivant la même approche que Hill (1987), Demko et Hill (1988) considèrent le paramètre  $\alpha \in [0, 1]$ , défini comme l'utilité maximum attribuée par un agent à un bien individuel. Ils montrent qu'il est possible d'allouer les biens indivisibles aux  $n$  agents de sorte que l'utilité d'un agent pour sa part vaille au moins  $V_n(\alpha)$ , donc  $t_n \geq V_n(\alpha)$ . De plus, Demko et Hill (1988) montrent que pour certaines instances,  $V_n(\alpha)$  est la meilleure fonction décroissante. En ce sens, il n'existe pas de fonction  $V'_n$  décroissante avec  $V'_n(\alpha) \geq V_n(\alpha)$  pour tout  $\alpha$  et  $V'_n(\alpha) > V_n(\alpha)$  pour certaines valeurs de  $\alpha$ , et pour laquelle chaque agent reçoit au moins  $V'_n(\alpha)$ . Nous montrerons qu'il existe une fonction  $W_n(\alpha) \geq V_n(\alpha)$  et que cette fonction  $W_n$  ne pourra pas être totalement décroissante. En fait, la fonction  $V_n$  alterne entre des phases strictement décroissantes sur les intervalles  $I(n, p)$  et des phases constantes sur les intervalles  $NI(n, p)$  pour tout entier  $p \geq 1$  (voir la définition 2.3 et la figure 2.1, page 63). Dans les phases strictement décroissantes,  $V_n$  est la meilleure borne inférieure de  $t_n$  (Demko et Hill, 1988). Ainsi,  $t_n = V_n$  sur  $I(n, p)$  pour tous les entiers  $n \geq 2$  et  $p \geq 1$ .

Bien que Demko et Hill (1988) s'intéressent également à l'utilité de l'agent le

plus défavorisé, leur approche est différente du critère égalitariste. Comme il n'y a que  $n$  et  $\alpha$  qui sont retenus, l'approche de Demko et Hill oublie partiellement l'instance et  $t_n$  peut être plus faible que la valeur qui optimise le critère égalitariste.

**Exemple 4.1** Observons les deux instances présentées dans le tableau 4.1.

Instance 1				Instance 2					
$e$	$e_1$	$e_2$	$e_3$	$e$	$e_1$	$e_2$	$e_3$	$e_4$	$e_5$
$u_1(e)$	0.4	0.3	0.3	$u_1(e)$	0.2	0.2	0.4	0.1	0.1
$u_2(e)$	1/3	1/3	1/3	$u_2(e)$	0.2	0.2	0.2	0.2	0.2

Tableau 4.1 – Exemple 4.1.

Dans ces deux instances,  $n = 2$  et  $\alpha$ , représentant l'utilité maximum d'un élément, vaut 0.4. L'optimum selon le critère égalitariste vaut 0.4 (en allouant  $\{e_1\}$  à l'agent 1 et  $\{e_2, e_3\}$  à l'agent 2) pour l'instance 1. Mais il est de 0.6 pour l'instance 2 (en allouant  $\{e_1, e_3\}$  à l'agent 1 et  $\{e_2, e_4, e_5\}$  à l'agent 2).

Ceci explique le paradoxe apparent d'avoir une 0.5-inapproximabilité pour le critère égalitariste et la possibilité de construire, en temps polynomial, une allocation où chaque agent  $i$  a au moins  $V_n(\alpha)$ .

En définissant  $\alpha_i$  comme la valeur maximum attribuée par l'agent  $i$  à un bien individuel, Markakis et Psomas (2011) ont renforcé le résultat de Demko et Hill. En effet, ils montrent l'existence d'une allocation garantissant  $V_n(\alpha_i)$  pour tout agent  $i$ . Puisque  $V_n$  est décroissante, alors  $V_n(\alpha_i) \geq V_n(\alpha)$  et le vecteur  $(V_n(\alpha_i))_{i \in [n]}$  Pareto-domine faiblement  $(V_n(\alpha))_{i \in [n]}$ . En d'autres termes, ils garantissent  $V_n(\alpha)$  à chacun et peut être plus pour les agents qui ne sont pas les plus défavorisés. L'autre contribution de Markakis et Psomas repose sur le fait que, contrairement au résultat de Demko et Hill (1988), l'allocation est obtenue avec un algorithme déterministe appelé ALLOCATE qui s'exécute en temps polynomial (voir l'algorithme 2.1, page 65).

Ce chapitre traite d'un problème qui généralise le problème classique d'allocation de biens indivisibles. Il s'agit de trouver une base d'un matroïde partagée par  $n$  agents. Les agents peuvent avoir différentes utilités pour les éléments du matroïde et ces utilités sont additives. Après normalisation, l'utilité maximum d'une base du matroïde est exactement 1 pour chaque agent. L'utilisation des matroïdes permet de couvrir des applications dont les contraintes sont plus complexes (voir la section 2.1.2 page 49).

Définissons  $\alpha_i$  comment étant l'utilité maximum qu'attribue l'agent  $i$  à un élément, et  $\alpha = \max_{i \in [n]} \alpha_i$ . Nous montrons que la garantie  $V_n$  reste valide dans

ce contexte généralisé et que nous pouvons même l'améliorer. Nous proposons une famille de fonctions  $W_n : [0, 1] \rightarrow [0, \frac{1}{n}]$  définies pour tout entier strictement positif  $n$ . Nous avons  $W_n(x) \geq V_n(x)$  pour tout  $x \in [0, 1]$ , et  $W_n(x) > V_n(x)$  pour les valeurs de  $x$  où  $V_n(x)$  n'est pas maximum (les phases constantes de  $V_n$ , c'est-à-dire les intervalles  $NI(n, p)$ ). Tout comme  $V_n$ ,  $W_n$  est linéaire et est définie par morceaux sur  $[0, 1]$  mais contrairement à  $V_n$ ,  $W_n$  alterne des phases strictement croissantes et strictement décroissantes.

Nous proposons également un algorithme déterministe qui est une extension de ALLOCATE de Markakis et Psomas (2011) (algorithme 2.1, page 65) pour le problème généralisé aux matroïdes. Notre algorithme retourne une solution (base-allocation) pour laquelle l'utilité de chaque agent  $i$  vaut au moins  $W_n(\alpha_i)$ . En présence de deux agents, une légère modification de cet algorithme permet d'obtenir une base dont l'utilité de chaque agent est au moins  $\max\{W_2(\alpha_1), W_2(\alpha_2)\}$ . Une borne supérieure sur l'utilité de l'agent le moins heureux est également proposée. Les bornes inférieure et supérieure proposées permettent de faire une estimation précise sur la valeur de  $t_n$ .

Notons que l'approche utilisée dans ce chapitre est équivalente à approcher le point idéal du problème multi-objectif associé à la base d'un matroïde. L'approximation du point idéal permet en effet de fournir une borne inférieure sur l'utilité de chaque agent pour la solution approchée, représentant notre *garantie dans le pire cas*. Cela est possible puisque les problématiques MA et MO définies dans la section 2.1 (page 43) sont équivalentes pour des matroïdes.

Le présent chapitre est organisé comme suit : la section 4.1 introduit le modèle que nous traitons. La section 4.1.1 présente les fonctions  $W_n$ . Dans la section 4.2, nous présentons un algorithme polynomial qui fournit une base-allocation d'un matroïde où l'utilité de chaque agent  $i$  vaut au moins  $W_n(\alpha_i)$ . Dans la section 4.3, nous présentons une légère modification de cet algorithme dans le cas où  $n = 2$  agents dans le but de garantir au moins  $\max\{W_2(\alpha_1), W_2(\alpha_2)\}$  à chacun. Dans la section 4.4, nous proposons une borne supérieure sur l'utilité de l'agent le plus défavorisé et nous fournissons une estimation du seuil  $t_n$  dans la section 4.4.1. Une conclusion et quelques perspectives de recherche sont discutées à la fin du chapitre.

## 4.1 Le modèle

Soient un matroïde  $\mathcal{M} = (X, \mathcal{F})$ , un ensemble d'agents  $N = \{1, \dots, n\}$  et des utilités  $u_i(e) \in \mathbb{R}^+$  pour chaque paire  $(i, e) \in N \times X$  où  $u_i(e)$  est l'utilité de l'agent  $i$  pour l'élément  $e$ . Par abus de notation, l'utilité d'un agent  $i \in N$  pour un sous-ensemble  $X'$  de  $X$  est notée  $u_i(X')$  et est définie par  $\sum_{x \in X'} u_i(x)$ . Par convention,  $u_i(\emptyset) = 0$ .

L'agent  $i$  préfère les solutions (bases) qui maximisent  $u_i$ . L'une de ces solutions, notée  $B_i^*$ , peut être construite par le GLOUTON (algorithme 1.1, page 32). Nous supposons, sans perte de généralité, que chaque base  $B_i^*$  optimale pour un agent  $i$  vérifie  $u_i(B_i^*) = 1$ . Cette normalisation permet d'obtenir des utilités relatives au lieu d'utilités absolues. Si  $u_i(B_i^*) \neq 1$  pour un certain agent  $i$ , alors il suffit de remplacer  $u_i(e)$  par  $\frac{u_i(e)}{u_i(B_i^*)}$  pour tout  $e \in X$ . Ceci est trivialement fait en temps polynomial.

Notons  $\alpha_i = \max_{\{e\} \in \mathcal{F}} u_i(e)$  l'utilité maximum de l'agent  $i$  pour un élément du matroïde et  $\alpha = \max_{i \in N} \alpha_i$  l'utilité maximum d'un élément du matroïde, parmi tous les agents.

En suivant l'approche de Demko et Hill (1988), notre objectif est de trouver une valeur  $t_n$  telle que, pour toute instance, il existe une base  $B \in \mathcal{F}$  avec  $B = \cup_{i \in N} B_i$  où les ensembles  $B_1, \dots, B_n$  sont deux à deux disjoints et tels que  $B_i$  représente la part de l'agent  $i$  et vérifie  $u_i(B) \geq u_i(B_i) \geq t_n$  pour tout  $i \in N$ . L'ensemble  $\{B_1, \dots, B_n\}$  forme donc une partition de  $B$ . Une telle solution est appelée *base-allocation*. Comme pour Markakis et Psomas (2011), un renforcement de ce résultat serait un vecteur  $(t_{i,n})_{i \in N}$  tel que, pour toute instance, il existe une solution (base-allocation)  $B = \{B_1, \dots, B_n\}$  du matroïde avec  $u_i(B_i) \geq t_{i,n} \geq t_n, \forall i \in N$ .

Puisque notre modèle étend les modèles étudiés par Demko et Hill (1988); Markakis et Psomas (2011), les paramètres  $\alpha$  et  $\alpha_i$  jouent un rôle important dans la détermination de  $t_n$  et  $(t_{i,n})_{i \in N}$ .

Rappelons que le rang d'un matroïde  $\mathcal{M} = (X, \mathcal{F})$ , noté  $r(\mathcal{M})$ , représente la taille des bases de  $\mathcal{M}$ . Par ailleurs, la *contraction*  $\mathcal{M}/F$  de  $\mathcal{M}$  par  $F \in \mathcal{F}$  est le matroïde  $(X \setminus F, \mathcal{F}')$  où  $\mathcal{F}' = \{E \subseteq X \setminus F : E \cup F \in \mathcal{F}\}$ . Notons par  $[n]$  l'ensemble  $\{1, \dots, n\}$ .

### 4.1.1 Les fonctions $W_n$

Les fonctions  $V_n$  sont présentées dans la section 2.3 (page 62). Nous définissons dans ce qui suit une nouvelle famille de fonctions  $W_n$ .

**Définition 4.1** *Étant donné un entier  $n \geq 1$ , soit  $W_n : [0, 1] \rightarrow [0, \frac{1}{n}]$  la fonction vérifiant  $W_1(x) = 1, \forall x \in [0, 1]$ , et pour tout entier  $n \geq 2$ ,  $W_n(x) = 1/n$  pour  $x = 0$  et pour  $x > 0$ ,*

$$W_n(x) = \begin{cases} 0, & x \in I_{(n,0)} \\ 1 - p(n-1)x, & x \in I_{(n,p)}^1 \\ \frac{p(1-px)}{(p+1)(n-1)-1}, & x \in I_{(n,p)}^2 \\ \frac{p(x+p-1)}{np^2-p-n+2}, & x \in I_{(n,p)}^3 \end{cases}$$

où  $I_{(n,0)} = \left[\frac{1}{n-1}, 1\right]$  et pour tout entier  $p \geq 1$ ,

$$I_{(n,p)} = \left[\frac{1}{(p+1)n-1}, \frac{1}{pn-1}\right] = I_{(n,p)}^1 \cup I_{(n,p)}^2 \cup I_{(n,p)}^3$$

avec

$$\begin{aligned} - I_{(n,p)}^1 &= \left[\frac{p+1}{p((p+1)n-1)}, \frac{1}{pn-1}\right], \\ - I_{(n,p)}^2 &= \left[\frac{p^2}{np^3-p^2+p+n-2}, \frac{p+1}{p((p+1)n-1)}\right], \\ - I_{(n,p)}^3 &= \left[\frac{1}{(p+1)n-1}, \frac{p^2}{np^3-p^2+p+n-2}\right]. \end{aligned}$$

Notons que  $I_{(n,p)} \neq I(n,p)$ . Dans le reste du chapitre, nous utilisons uniquement les intervalles de la définition 4.1. La figure 4.1 contient une représentation graphique de  $W_n$  comparée à  $V_n$  pour  $n = 2$  et  $n = 3$ .

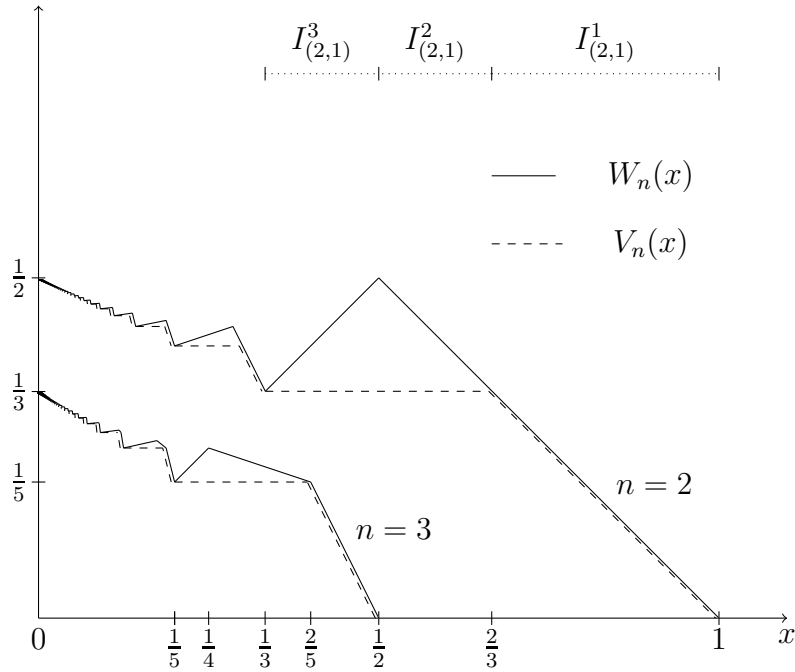


FIGURE 4.1 –  $V_n$  et  $W_n$  pour  $n = 2$  et  $n = 3$ .

Les fonctions  $W_n$  sont non-monotones et sont définies par des segments délimités par les intervalles  $I_{(n,p)}^1$ ,  $I_{(n,p)}^2$  et  $I_{(n,p)}^3$  pour des entiers  $n \geq 2$  et  $p \geq 1$ . Les segments de  $I_{(n,p)}^1$  et  $I_{(n,p)}^2$  sont strictement décroissants et ceux de  $I_{(n,p)}^3$  sont

strictement croissants. À titre d'exemple, pour  $n = 2$  et  $p = 1$ , on a

$$W_2(x) = \begin{cases} 1 - x, & x \in I_{(2,1)}^1 = \left[\frac{2}{3}, 1\right[ \\ 1 - x, & x \in I_{(2,1)}^2 = \left[\frac{1}{2}, \frac{2}{3}\right[ \\ x, & x \in I_{(2,1)}^3 = \left[\frac{1}{3}, \frac{1}{2}\right[ \end{cases}$$

Nous présentons des propriétés sur les fonctions  $W_n$  dont les preuves sont dans l'annexe B, page 179.

**Propriété 4.1** *Étant donnés des entiers  $n \geq 2$  et  $p \geq 1$ , alors pour tout  $x \in [0, 1]$ ,  $V_n(x) \leq W_n(x) \leq \frac{1}{n}$ .*

**Propriété 4.2** *Étant donné un entier  $n \geq 2$ , si  $x \leq \frac{p+1}{p((p+1)n-1)}$  pour un certain entier  $p \geq 1$ , alors  $W_n(x) \geq \frac{p}{(p+1)n-1}$ .*

**Propriété 4.3** *Étant donné un entier  $n \geq 2$ , si  $x \in I_{(n,p)}$  pour un certain entier  $p \geq 1$ , alors  $(p-1)x < W_n(x) \leq px$ .*

## 4.2 L'algorithme SEUIL

Nous présentons un algorithme polynomial appelé SEUIL (voir l'algorithme 4.1) qui construit une base d'un matroïde commune à  $n$  agents. Pour cela, on exécute  $\text{SEUIL}(N, \mathcal{M}, (u_i)_{i \in N}, \emptyset)$ .

L'algorithme SEUIL est une adaptation d'ALLOCATE de Markakis et Psomas (2011) (algorithme 2.1, page 65) aux matroïdes. Il s'inspire également de l'algorithme 3.3 (page 85) utilisé dans la construction d'une base-allocation proche de la proportionnalité (voir la section 3.4, page 83).

L'algorithme SEUIL est récursif en  $n$ . Chaque itération consiste à choisir un agent  $i$  qui ajoute des éléments à la solution, dans le but de se garantir  $W_n(\alpha_i)$ . La sélection d'un agent est faite de la façon suivante : chaque agent  $k \in N$  atteint la garantie  $W_n(\alpha_k)$  avec un nombre minimum d'éléments. Puis on choisit l'agent qui utilise le plus petit nombre d'éléments. Ceci dans le but d'assurer le plus grand nombre d'éléments restants à la disposition des autres agents dans les prochaines itérations de l'algorithme. L'union des éléments de toutes les itérations forme une base du matroïde.

La complexité de l'algorithme SEUIL est majorée par les  $O(n^2)$  exécutions du GLOUTON (algorithme 1.1, page 32). La complexité de SEUIL est par conséquent  $O(n^2 |X| \max\{\ln |X|, \theta\})$ .



---

**Algorithme 4.1 : SEUIL**

---

**Données :** un ensemble d'agents  $N$ , un matroïde  $\mathcal{M} = (X, \mathcal{F})$ , des utilités  $(u_i)_{i \in N}$ , une solution partielle  $B$

**Résultat :** une base-allocation  $B$

```

1 pour tout  $i \in N$  faire
2    $B_i^* \leftarrow \text{GLOUTON}(\mathcal{M}, u_i, \max)$ 
3   pour tout  $e \in X$  faire
4      $\tilde{u}_i(e) \leftarrow \frac{u_i(e)}{u_i(B_i^*)}$ 
5    $\alpha_i \leftarrow \max_{\{e\} \in \mathcal{F}} \tilde{u}_i(e)$ 
6    $S_i \leftarrow \emptyset$ 
7   ajouter de façon gloutonne des éléments de  $B_i^*$  à  $S_i$  par ordre décroissant de  $\tilde{u}_i$  jusqu'à
   ce que  $\tilde{u}_i(S_i) \geq W_{|N|}(\alpha_i)$ 
8 choisir un agent  $i \in N$  tel que  $|S_i| \leq |S_k|, \forall k \in N$ 
9  $B_i \leftarrow S_i$ 
10  $B \leftarrow B \cup B_i$ 
11 si  $|N| = 1$  alors
12   retourner  $B$ 
13 sinon
14   soit  $\mathcal{M}/B_i = (X \setminus B_i, \tilde{\mathcal{F}})$  la contraction de  $\mathcal{M}$  à  $B_i$  où  $F \in \tilde{\mathcal{F}}$  si et seulement si
    $F \cup B_i \in \mathcal{F}$ 
15   SEUIL  $(N \setminus \{i\}, \mathcal{M}/B_i, (\tilde{u}_k)_{k \in N \setminus \{i\}}, B)$ 

```

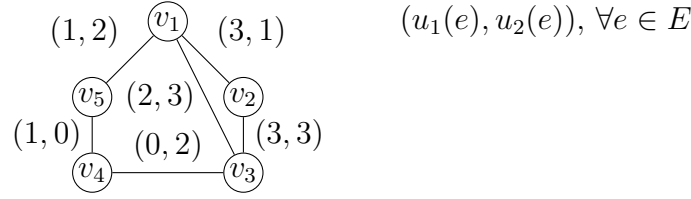
---

Dans le reste du chapitre, nous supposons toujours que l'agent  $i$  est sélectionné durant le  $i^{\text{ème}}$  appel de SEUIL. Ainsi, à la fin du  $i^{\text{ème}}$  appel récursif,  $B = B_1 \cup \dots \cup B_i$  et le matroïde contracté est  $\mathcal{M}/(B_1 \cup \dots \cup B_i)$ . À la fin du  $n^{\text{ème}}$  appel de SEUIL,  $B = B_1 \cup \dots \cup B_n$  est une base de  $\mathcal{M}$  car durant le  $n^{\text{ème}}$  appel, l'algorithme ajoute un ensemble indépendant qui est par construction une base de  $\mathcal{M}/(B_1 \cup \dots \cup B_{n-1})$ .

Considérons les exemples suivants sur l'application de l'algorithme SEUIL.

**Exemple 4.2** Soient un ensemble d'agents  $N = \{1, 2\}$  et le matroïde graphique défini sur le graphe connexe  $G = (V, E)$ , représenté dans la figure 4.2. L'objectif est de trouver une base-allocation pour les deux agents avec une garantie sur l'utilité de chacun. Dans le matroïde graphique de  $G$ , une base correspond à un arbre couvrant. Nous utilisons le matroïde graphique afin de mieux illustrer les notions de base, ensemble indépendant, contraction, etc.

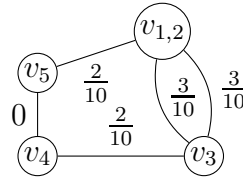
Durant le premier appel de SEUIL,  $B_1^* = \{(v_1, v_2), (v_2, v_3), (v_4, v_5), (v_5, v_1)\}$  et  $B_2^* = \{(v_2, v_3), (v_1, v_3), (v_3, v_4), (v_5, v_1)\}$  avec  $u_1(B_1^*) = 8$  et  $u_2(B_2^*) = 10$  où  $B_i^*$  correspond à un arbre couvrant maximum dans  $(G, u_i)$  (par exemple, en appliquant le GLOUTON ou l'algorithme de Kruskal). L'algorithme SEUIL normalise les utilités  $u_i(e)$  afin d'obtenir  $\tilde{u}_i(e) = \frac{u_i(e)}{u_i(B_i^*)}, \forall i \in \{1, 2\}, \forall e \in E$ . Alors,  $\alpha_1 = \frac{3}{8} \in I_{(2,1)}^3$  et


 FIGURE 4.2 – Exemple 4.2 : le graphe  $G$ .

$\alpha_2 = \frac{3}{10} \in I_{(2,2)}^1$ . Ainsi,  $W_2(\alpha_1) = \alpha_1 = \frac{3}{8}$  et  $W_2(\alpha_2) = 1 - 2\alpha_2 = \frac{4}{10}$ .

Chaque agent  $i \in \{1, 2\}$  construit une forêt  $S_i$  en ajoutant les éléments de poids maximum de  $B_i^*$  jusqu'à ce que  $\tilde{u}_i(S_i) \geq W_2(\alpha_i)$ . On obtient  $S_1 = \{(v_1, v_2)\}$  et  $S_2 = \{(v_2, v_3), (v_1, v_3)\}$ . Puisque  $|S_1| < |S_2|$ , alors l'agent  $i = 1$  est choisi, sa part est  $B_1 = S_1$  et la solution partielle est  $B = B_1$ .

Soit  $G/B_1$  le nouveau graphe obtenu en contractant l'arête  $(v_1, v_2)$  de  $B_1$  en un sommet  $v_{1,2}$  comme défini dans la figure 4.3.


 FIGURE 4.3 – Exemple 4.2 : le graphe contracté  $G/B_1$ .

Les utilités de l'agent 1 sont supprimées car cet agent a déjà apporté sa contribution. Rappelons que l'utilité du second agent a été normalisée durant le premier appel récursif de SEUIL :  $\tilde{u}_2(e) = \frac{u_2(e)}{u_2(B_2^*)}$ . Soit  $\tilde{B}_2^*$  une base optimale du matroïde contracté  $(G/B_1, \tilde{u}_2)$ . Durant le second appel de SEUIL, l'algorithme trouve  $\tilde{B}_2^* = \{(v_{1,2}, v_3), (v_3, v_4), (v_5, v_{1,2})\}$  avec  $\tilde{u}_2(\tilde{B}_2^*) = \frac{7}{10}$ . Les utilités de l'agent 2 sont maintenant normalisées  $\tilde{\tilde{u}}_2(e) = \frac{\tilde{u}_2(e)}{\tilde{u}_2(\tilde{B}_2^*)}$  et  $\tilde{\alpha}_2 = \frac{3}{10} * \frac{10}{7} = \frac{3}{7}$ . Alors, l'agent 2 construit une forêt  $\tilde{S}_2$  en ajoutant les éléments de poids maximum de  $\tilde{B}_2^*$  jusqu'à atteindre le seuil  $\tilde{\tilde{u}}_2(\tilde{S}_2) \geq W_1(\tilde{\alpha}_2) = 1$ . Donc  $B_2 = \tilde{B}_2^*$ .

Finalement, l'algorithme SEUIL retourne la base-allocation  $B = B_1 \cup B_2$  où  $B_1 = \{(v_1, v_2)\}$  et  $B_2 = \{(v_2, v_3), (v_3, v_4), (v_5, v_1)\}$  avec  $u_1(B_1) = 3 = W_2(\alpha_1)u_1(B_1^*)$  et  $u_2(B_2) = 7 > W_2(\alpha_2)u_2(B_2^*) = 4$ .

Le prochain exemple montre que la garantie  $W_n$  est meilleure que  $V_n$ .

**Exemple 4.3** Considérons le matroïde de partition  $\mathcal{M} = (X, \mathcal{F})$  défini par  $X = \cup_{i \in [4]} X_i$  où  $X_i = \{e_i^1, e_i^2\}$  pour tout  $i \in [4]$ . Un ensemble  $F$  appartient à  $\mathcal{F}$  si

$F \subseteq X$  et  $|F \cap X_i| \leq 1$  pour tout  $i \in [4]$ . Ce matroïde modélise le problème classique d'allocation de biens indivisibles avec 4 objets et 2 agents. L'objectif est d'obtenir une base qui correspond à une allocation. En effet, si un élément  $e_i^j$  est sélectionné dans la base, alors le bien  $i$  est alloué à l'agent  $j$ , pour  $i \in \{1, \dots, 4\}$  et  $j \in \{1, 2\}$  (voir la définition du matroïde-allocation en page 51). Les utilités des agents sont présentées dans le tableau 4.2.

	$X_1$		$X_2$		$X_3$		$X_4$	
$e$	$e_1^1$	$e_1^2$	$e_2^1$	$e_2^2$	$e_3^1$	$e_3^2$	$e_4^1$	$e_4^2$
$u_1(e)$	0.4	0	0.2	0	0.2	0	0.2	0
$u_2(e)$	0	0.35	0	0.3	0	0.3	0	0.05

Tableau 4.2 – Exemple 4.3.

La base  $B_j^* = \{e_1^j, e_2^j, e_3^j, e_4^j\}$  est optimale pour l'agent  $j$  avec  $u_j(B_j^*) = 1$  pour  $j = 1, 2$ , donc l'instance est normalisée à 1. Par conséquent,  $\alpha_1 = 0.4 \in I_{(2,1)}^3$  donc  $W_2(0.4) = 0.4$ , et  $\alpha_2 = 0.35 \in I_{(2,1)}^3$  donc  $W_2(0.35) = 0.35$ .

L'exécution de SEUIL ( $\{1, 2\}, \mathcal{M}, (u_1, u_2), \emptyset$ ) retourne la base  $B = B_1 \cup B_2$  où  $B_1 = \{e_1^1\}$  et  $B_2 = \{e_2^2, e_3^2, e_4^2\}$  avec  $u_1(B) = u_1(B_1) = 0.4 = W_2(0.4) = 0.4 > V_2(0.4) = \frac{1}{3}$  et  $u_2(B) = u_2(B_2) = 0.65 > W_2(0.35) = 0.35 > V_2(0.35) = \frac{1}{3}$ .

Remarquons que le matroïde est construit de façon à ce que  $u_i(B) = u_i(B_i)$ ,  $i = 1, 2$ . D'un point de vue MA, l'utilité d'un agent  $i$  est évaluée sur sa part  $B_i$ . Or, le problème MA peut être réduit à un problème MO où l'utilité de chaque agent est évaluée sur la solution  $B$  dans son intégralité. Ceci est possible par construction du matroïde-allocation (voir la section 2.1.2, page 49).

Dans ce qui suit, nous fournissons des garanties sur  $u_i(B)$  pour tout  $i \in N$  mais ces garanties sont également valables sur  $u_i(B_i)$ .

Le prochain théorème représente le résultat principal de ce chapitre et fournit une garantie à l'algorithme SEUIL pour obtenir une base du problème.

**Théorème 4.1** *L'algorithme SEUIL retourne une base  $B$  vérifiant*

$$u_i(B) \geq W_{|N|}(\alpha_i)u_i(B_i^*), \quad \forall i \in N$$

**Preuve.** Soit  $n = |N|$  et rappelons que la base retournée par SEUIL est  $B = B_1 \cup \dots \cup B_n$ . Nous prouvons un résultat plus fort que ce qui est donné dans l'énoncé par récurrence sur  $n$  :  $u_i(B_i) \geq W_n(\alpha_i)u_i(B_i^*)$ ,  $\forall i \in N$ . Puisque les utilités sont positives, nous pouvons conclure que  $u_i(B) \geq u_i(B_i)$ .

Pour  $n = 1$ , la ligne 7 de SEUIL est équivalente à appliquer GLOUTON  $(\mathcal{M}, u_1, \max)$  afin d'obtenir la base  $B = B_1 = B_1^*$  qui vérifie :  $\tilde{u}_1(B_1) \geq W_1(\alpha_1) = 1$  car  $W_1(x) = 1, \forall x \in [0, 1]$ . Ainsi,  $u_1(B_1) = W_1(\alpha_1)u_1(B_1^*)$ .

Soit  $n \geq 2$ . Nous supposons que  $r(\mathcal{M}) \geq n$  car autrement toutes les bases contiendraient au plus  $n - 1$  éléments et par conséquent,  $\alpha_i \geq 1/(n - 1)$  pour tout  $i \in N$ . Dans ce cas,  $W_n(\alpha_i) = 0$  pour tout  $i \in N$  et le résultat est obtenu trivialement.

Pour des raisons similaires, nous supposons que  $\alpha_i < 1/(n - 1), \forall i \in N$  car autrement pour tout agent  $i$  avec  $\alpha_i \geq 1/(n - 1)$ , on obtiendrait  $W_n(\alpha_i) = 0$  et la borne de cet agent  $i$  est clairement satisfaite.

Sans perte de généralité, nous supposons que l'agent 1 est sélectionné en premier, donc  $B_1 = S_1$  et on déduit grâce à la ligne 8 de SEUIL que

$$|B_1| \leq |S_k|, \forall k \in N \quad (4.1)$$

Afin d'éviter toute confusion entre les notations du premier et du deuxième appel récursif de l'algorithme SEUIL, nous ajoutons un tilde pour les notations utilisées durant le deuxième appel de SEUIL (comme dans l'exemple 4.2, page 104). Ainsi,  $\tilde{N} = N \setminus \{1\}$ ,  $\tilde{u}_i(e) = u_i(e)/u_i(B_i^*)$ ,  $\tilde{\mathcal{M}} = \mathcal{M}/B_1 = (X \setminus B_1, \tilde{\mathcal{F}})$  et  $\tilde{B}_i^*$  est une base optimale de  $(\tilde{\mathcal{M}}, \tilde{u}_i)$ . De plus, pour  $i \in \tilde{N}$ ,  $\tilde{u}_i(e) = \tilde{u}_i(e)/\tilde{u}_i(\tilde{B}_i^*)$  pour les éléments  $e \in X \setminus B_1$  de  $\tilde{\mathcal{M}}$  et  $\tilde{\alpha}_i = \max_{\{e\} \in \tilde{\mathcal{F}}} \tilde{u}_i(e)$ .

L'hypothèse de récurrence affirme que

$$\tilde{u}_i(B_i) \geq W_{n-1}(\tilde{\alpha}_i)\tilde{u}_i(\tilde{B}_i^*), \forall i \in \tilde{N} \quad (4.2)$$

Nous voulons montrer que  $u_i(B_i) \geq W_n(\alpha_i)u_i(B_i^*), \forall i \in N$  ou de façon équivalente,  $\tilde{u}_i(B_i) \geq W_n(\alpha_i), \forall i \in N$ . La construction de l'algorithme SEUIL permet de conclure que durant le premier appel (lignes 3-7), on a

$$\tilde{u}_i(S_i) \geq W_n(\alpha_i), \forall i \in N \quad (4.3)$$

Puisque  $B_1 = S_1$ , alors en utilisant l'inégalité (4.3), on obtient le résultat attendu pour l'agent 1. De plus, les ensembles  $S_i = \{e_i^1, \dots, e_i^{|S_i|}\}$  sont minimaux pour l'inclusion car les éléments de  $S_i$  sont ajoutés graduellement. Ainsi,

$$\tilde{u}_i(S_i \setminus \{e_i^{|S_i|}\}) < W_n(\alpha_i), \forall i \in N \quad (4.4)$$

Soit  $k \in \tilde{N}$ . Nous partitionnons  $B_k^*$  en  $B_k^{*1}$  et  $B_k^{*2}$  de façon à ce que  $B_k^{*1}$  contienne les  $|B_1|$  premiers éléments de poids maximum de  $B_k^*$ . Ceci est possible grâce à (4.1) et en rappelant que  $S_k \subseteq B_k^*$  (voir la ligne 7 de SEUIL). D'où,  $B_k^{*1} \subseteq S_k$  et

$$\tilde{u}_k(B_k^{*1}) \leq \tilde{u}_k(S_k) \quad (4.5)$$

Puisque pour  $n \geq 2$ ,  $W_n(x) \leq 1/n \leq 1/2$  (voir la propriété 4.1, page 103) et  $r(\mathcal{M}) \geq n \geq 2$ , nous pouvons prouver que  $|B_1| < |B_1^*| = |B_k^*|$  (toutes les bases sont de même cardinalité  $r(\mathcal{M})$ ). Montrons-le par contradiction : supposons que  $|B_1| = |B_1^*|$ . Puisque  $S_1 = B_1$  et  $S_1 \subseteq B_1^*$ , alors  $S_1 = B_1 = B_1^*$ , donc  $\tilde{u}_1(S_1) = \tilde{u}_1(B_1^*) = 1$ . En utilisant (4.4),  $\tilde{u}_1(S_1 \setminus \{e_1^{|S_1|}\}) = \tilde{u}_1(S_1) - \tilde{u}_1(e_1^{|S_1|}) = 1 - \tilde{u}_1(e_1^{|S_1|}) < W_n(\alpha_1) \leq 1/2$ , donc  $\tilde{u}_1(e_1^{|S_1|}) > 1/2$ . Puisque  $|S_1| = r(\mathcal{M}) \geq 2$  et  $S_1$  est trié par ordre décroissant de  $\tilde{u}_1$ , alors  $\tilde{u}_1(S_1) \geq 2\tilde{u}_1(e_1^{|S_1|}) > 1$  qui contredit l'hypothèse de normalisation.

Nous pouvons alors appliquer le lemme 1.2 (page 33) pour  $\mathcal{M}$ ,  $w = \tilde{u}_k$ ,  $F_1 = B_1$  et  $F_2 = B_k^*$ . Nous pouvons conclure qu'il existe  $E \subseteq B_k^* \setminus B_1$  tel que  $B_1 \cup E \in \mathcal{F}$  où  $|E| = |B_k^*| - |B_1| = |B_k^{*2}|$  et  $\tilde{u}_k(E) \geq \tilde{u}_k(B_k^{*2})$ .  $E$  est un ensemble indépendant de  $\tilde{\mathcal{M}} = \mathcal{M}/B_1$  qui est maximum pour  $\tilde{u}_k$ , tandis que  $\tilde{B}_k^*$  est une base de  $\tilde{\mathcal{M}}$  maximum pour  $\tilde{u}_k$ . D'où,  $\tilde{u}_k(\tilde{B}_k^*) \geq \tilde{u}_k(E) \geq \tilde{u}_k(B_k^{*2})$ . Puisque l'instance est normalisée à 1, alors

$$\tilde{u}_k(\tilde{B}_k^*) \geq 1 - \tilde{u}_k(B_k^{*1}) \quad (4.6)$$

De (4.5) et (4.6), on déduit que

$$\tilde{u}_k(\tilde{B}_k^*) \geq 1 - \tilde{u}_k(S_k) \quad (4.7)$$

et de (4.2) et (4.7), on obtient

$$\tilde{u}_k(B_k) \geq W_{n-1}(\tilde{\alpha}_k) (1 - \tilde{u}_k(S_k)) \quad (4.8)$$

Nous savons grâce à la ligne 5 de SEUIL que

$$\tilde{\alpha}_k = \max_{B_1 \cup \{e\} \in \mathcal{F}} \tilde{u}_k(e) = \frac{\max_{B_1 \cup \{e\} \in \mathcal{F}} \tilde{u}_k(e)}{\tilde{u}_k(\tilde{B}_k^*)} \leq \frac{\alpha_k}{\tilde{u}_k(\tilde{B}_k^*)}$$

En utilisant (4.7), la dernière inégalité devient

$$\tilde{\alpha}_k \leq \frac{\alpha_k}{1 - \tilde{u}_k(S_k)} \quad (4.9)$$

Soit  $p_k \in \mathbb{N}^*$  tel que  $\alpha_k \in I_{(n,p_k)}$  (voir la définition 4.1, page 101).

Commençons par montrer que  $|S_k| \geq p_k$ . Par contradiction, supposons que  $|S_k| \leq p_k - 1$ . La définition de  $\alpha_k$  permet de déduire que  $\tilde{u}_k(S_k) \leq (p_k - 1)\alpha_k$ . En utilisant la propriété 4.3 (page 103), on obtient  $(p_k - 1)\alpha_k < W_n(\alpha_k)$  qui contredit (4.3). Ainsi,  $|S_k| \geq p_k$ .

On distingue deux cas :  $|S_k| = p_k$  et  $|S_k| \geq p_k + 1$ .

**Cas 1 :**  $|S_k| = p_k$ .

La ligne 5 de SEUIL implique que  $\tilde{u}_k(S_k) \leq p_k \alpha_k$ . L'inégalité (4.8) devient

$$\tilde{u}_k(B_k) \geq W_{n-1}(\tilde{\alpha}_k) (1 - p_k \alpha_k) \quad (4.10)$$

et de (4.9), on déduit que

$$\begin{aligned}\tilde{\alpha}_k &\leq \frac{\alpha_k}{1 - p_k \alpha_k} \\ &< \frac{1}{p_k(n-1) - 1}\end{aligned}\quad (4.11)$$

car  $\alpha_k < \frac{1}{p_k n - 1}$  (rappelons que  $\alpha_k \in I_{(n, p_k)}$ ). Ainsi,  $\tilde{\alpha}_k \in I_{(n-1, p_k)}^1$  ou  $\tilde{\alpha}_k \leq \frac{p_k + 1}{p_k((p_k + 1)(n-1) - 1)}$ .

(a) Si  $\tilde{\alpha}_k \in I_{(n-1, p_k)}^1$  alors  $W_{n-1}(\tilde{\alpha}_k) = 1 - p_k(n-2)\tilde{\alpha}_k$ . De (4.10), on déduit que  $\tilde{u}_k(B_k) \geq (1 - p_k(n-2)\tilde{\alpha}_k)(1 - p_k \alpha_k)$ . En utilisant (4.11), on obtient

$$\tilde{u}_k(B_k) \geq \left(1 - p_k(n-2) \frac{\alpha_k}{1 - p_k \alpha_k}\right) (1 - p_k \alpha_k) = 1 - p_k(n-1)\alpha_k$$

Considérons les différentes valeurs de  $\alpha_k \in I_{(n, p_k)} = I_{(n, p_k)}^1 \cup I_{(n, p_k)}^2 \cup I_{(n, p_k)}^3$  :

- si  $\alpha_k \in I_{(n, p_k)}^1$  alors  $\tilde{u}_k(B_k) \geq 1 - p_k(n-1)\alpha_k = W_n(\alpha_k)$ ,
- si  $\alpha_k \in I_{(n, p_k)}^2$  alors  $\tilde{u}_k(B_k) \geq 1 - p_k(n-1)\alpha_k > \frac{p_k(1 - p_k \alpha_k)}{(p_k + 1)(n-1) - 1} = W_n(\alpha_k)$  car  $\alpha_k < \frac{p_k + 1}{p_k((p_k + 1)n - 1)}$ ,
- si  $\alpha_k \in I_{(n, p_k)}^3$  alors  $\tilde{u}_k(B_k) \geq 1 - p_k(n-1)\alpha_k > \frac{p_k(\alpha_k + p_k - 1)}{np_k^2 - p_k - n + 2} = W_n(\alpha_k)$  si et seulement si  $\alpha_k < \frac{(n-1)p_k^2 - n + 2}{p_k(n(n-1)p_k^2 - (n-1)p_k - n^2 + 3n - 1)}$ .

Par hypothèse,  $\alpha_k < \frac{p_k^2}{np_k^3 - p_k^2 + p_k + n - 2}$  (car  $\alpha_k \in I_{(n, p_k)}^3$ ), alors  $\frac{p_k^2}{np_k^3 - p_k^2 + p_k + n - 2} \leq \frac{(n-1)p_k^2 - n + 2}{p_k(n(n-1)p_k^2 - (n-1)p_k - n^2 + 3n - 1)}$  si et seulement si  $np_k^2 - p_k - n + 2 \geq 0$  qui est vrai puisque  $n \geq 2$  et  $p_k \geq 1$ .

(b) Sinon,  $\tilde{\alpha}_k \leq \frac{p_k + 1}{p_k((p_k + 1)(n-1) - 1)}$ . En utilisant la propriété 4.2 (page 103) si  $n \geq 3$  ou  $W_1(x) = 1$  si  $n = 2$ , on obtient  $W_{n-1}(\tilde{\alpha}_k) \geq \frac{p_k}{(p_k + 1)(n-1) - 1}$ . De (4.10), on obtient

$$\tilde{u}_k(B_k) \geq \frac{p_k}{(p_k + 1)(n-1) - 1} (1 - p_k \alpha_k)$$

Considérons les différentes valeurs de  $\alpha_k \in I_{(n, p_k)} = I_{(n, p_k)}^1 \cup I_{(n, p_k)}^2 \cup I_{(n, p_k)}^3$  :

- si  $\alpha_k \in I_{(n, p_k)}^1$  alors  $\tilde{u}_k(B_k) \geq \frac{p_k(1 - p_k \alpha_k)}{(p_k + 1)(n-1) - 1} \geq 1 - p_k(n-1)\alpha_k = W_n(\alpha_k)$  car  $\alpha_k \geq \frac{p_k + 1}{p_k((p_k + 1)n - 1)}$ ,
- si  $\alpha_k \in I_{(n, p_k)}^2$  alors  $\tilde{u}_k(B_k) \geq \frac{p_k(1 - p_k \alpha_k)}{(p_k + 1)(n-1) - 1} = W_n(\alpha_k)$ ,
- si  $\alpha_k \in I_{(n, p_k)}^3$  alors  $\tilde{u}_k(B_k) \geq \frac{p_k(1 - p_k \alpha_k)}{(p_k + 1)(n-1) - 1} > \frac{p_k(\alpha_k + p_k - 1)}{np_k^2 - p_k - n + 2} = W_n(\alpha_k)$  car  $\alpha_k < \frac{p_k^2}{np_k^3 - p_k^2 + p_k + n - 2}$ .

**Cas 2 :**  $|S_k| \geq p_k + 1$ .

Montrons que  $p_k \geq 2$ . Par contradiction, supposons que  $p_k = 1$ , alors  $|S_k| \geq 2$ . Puisque  $n \geq 2$ , alors en utilisant la propriété 4.3 (page 103) avec  $\alpha_k \in I_{(n,p_k)}$ , on obtient  $W_n(\alpha_k) \leq p_k \alpha_k = \alpha_k$  car  $p_k = 1$ . Ainsi, l'agent  $k$  atteint le seuil  $\tilde{u}_k(S_k) = \alpha_k \geq W_n(\alpha_k)$  en sélectionnant l'élément de poids maximum de  $B_k^*$ , ce qui contredit le fait que  $|S_k| \geq 2$ .

En utilisant l'inégalité (4.4), on obtient  $W_n(\alpha_k) > \tilde{u}_k(S_k \setminus \{e_k^{|S_k|}\}) \geq \tilde{u}_k(e_k^1) + (p_k - 1)\tilde{u}_k(e_k^{p_k}) = \alpha_k + (p_k - 1)\tilde{u}_k(e_k^{p_k})$  pour les raisons suivantes :  $|S_k| \geq p_k + 1$ , les éléments de  $S_k$  sont triés par ordre décroissant de  $\tilde{u}_k$  et  $\tilde{u}_k(e_k^1) = \alpha_k$ . De cette dernière inégalité, on déduit que  $\tilde{u}_k(e_k^{p_k}) < \frac{W_n(\alpha_k) - \alpha_k}{p_k - 1}$  car  $p_k \geq 2$ , alors  $\tilde{u}_k(e_k^{|S_k|}) \leq \tilde{u}_k(e_k^{p_k}) < \frac{W_n(\alpha_k) - \alpha_k}{p_k - 1}$  (car  $S_k$  est trié par ordre décroissant de  $\tilde{u}_k$ ). Finalement, en ajoutant la dernière inégalité de  $\tilde{u}_k(e_k^{|S_k|})$  à (4.4), on obtient

$$\tilde{u}_k(S_k) < \frac{p_k W_n(\alpha_k) - \alpha_k}{p_k - 1} \quad (4.12)$$

D'une part, de (4.8) et (4.12), on a

$$\tilde{u}_k(B_k) \geq W_{n-1}(\tilde{\alpha}_k) \left( 1 - \frac{p_k W_n(\alpha_k) - \alpha_k}{p_k - 1} \right) \quad (4.13)$$

D'autre part, de (4.9) et (4.12), on obtient

$$\tilde{\alpha}_k \leq \frac{\alpha_k(p_k - 1)}{p_k - 1 + \alpha_k - p_k W_n(\alpha_k)} \quad (4.14)$$

Analysons les différentes valeurs de  $\alpha_k$  dans  $I_{(n,p_k)} = I_{(n,p_k)}^1 \cup I_{(n,p_k)}^2 \cup I_{(n,p_k)}^3$ .

**(a) :**  $\alpha_k \in I_{(n,p_k)}^1$ . Par construction,  $W_n(\alpha_k) = 1 - p_k(n-1)\alpha_k$ . En utilisant (4.13), on obtient

$$\tilde{u}_k(B_k) \geq W_{n-1}(\tilde{\alpha}_k) \frac{((n-1)p_k^2 + 1)\alpha_k - 1}{p_k - 1} \quad (4.15)$$

Par ailleurs, de (4.14) on obtient  $\tilde{\alpha}_k \leq \frac{\alpha_k(p_k - 1)}{((n-1)p_k^2 + 1)\alpha_k - 1}$ .

Montrons que  $\tilde{\alpha}_k \leq \frac{p_k + 1}{p_k((p_k + 1)(n-1) - 1)}$ . L'inégalité  $\frac{\alpha_k(p_k - 1)}{((n-1)p_k^2 + 1)\alpha_k - 1} \leq \frac{p_k + 1}{p_k((p_k + 1)(n-1) - 1)}$  est valide si et seulement si  $\alpha_k \geq \frac{p_k + 1}{np_k^2 + (n-1)p_k + 1}$ . Puisque  $\alpha_k \in I_{(n,p_k)}^1$ , il s'ensuit que  $\alpha_k \geq \frac{p_k + 1}{p_k((p_k + 1)n - 1)} \geq \frac{p_k + 1}{np_k^2 + (n-1)p_k + 1}$  est toujours vrai. D'où,

$$\tilde{\alpha}_k \leq \frac{p_k + 1}{p_k((p_k + 1)(n-1) - 1)} \quad (4.16)$$

En utilisant la propriété 4.2 (page 103) si  $n \geq 3$  avec (4.16) ou  $W_1(x) = 1$  si  $n = 2$ , on déduit que  $W_{n-1}(\tilde{\alpha}_k) \geq \frac{p_k}{(p_k + 1)(n-1) - 1}$ . L'inégalité (4.15) devient

$\tilde{u}_k(B_k) \geq \left( \frac{p_k}{(p_k+1)(n-1)-1} \right) \frac{((n-1)p_k^2+1)\alpha_k-1}{p_k-1} \geq 1-p_k(n-1)\alpha_k = W_n(\alpha_k)$  si et seulement si  $\alpha_k \geq \frac{(n-1)p_k^2-n+2}{p_k(n(n-1)p_k^2-(n-1)p_k-n^2+3n-1)}$ , mais  $\alpha_k \in I_{(n,p_k)}^1$ , donc  $\alpha_k \geq \frac{p_k+1}{p_k((p_k+1)n-1)} \geq \frac{(n-1)p_k^2-n+2}{p_k(n(n-1)p_k^2-(n-1)p_k-n^2+3n-1)}$  car  $n, p_k \geq 2$ .

**(b) :**  $\alpha_k \in I_{(n,p_k)}^2$ . Par construction,  $W_n(\alpha_k) = \frac{p_k(1-p_k\alpha_k)}{(p_k+1)(n-1)-1}$ . D'une part, de (4.13),

$$\tilde{u}_k(B_k) \geq W_{n-1}(\tilde{\alpha}_k) \frac{(p_k^3 + (n-1)p_k + n-2)\alpha_k + (n-2)p_k^2 - p_k - n + 2}{(p_k-1)((p_k+1)(n-1)-1)} \quad (4.17)$$

D'autre part, de (4.14),

$$\tilde{\alpha}_k \leq \frac{(p_k-1)((p_k+1)(n-1)-1)\alpha_k}{(p_k^3 + (n-1)p_k + n-2)\alpha_k + (n-2)p_k^2 - p_k - n + 2} \quad (4.18)$$

Montrons que  $\tilde{\alpha}_k \leq \frac{p_k+1}{p_k((p_k+1)(n-1)-1)}$ .

Si  $n = 2$  alors (4.18) donne  $\tilde{\alpha}_k \leq \frac{(p_k-1)\alpha_k}{(p_k^2+1)\alpha_k-1} \leq \frac{p_k+1}{p_k^2}$  si et seulement si  $\alpha_k \geq \frac{p_k+1}{2p_k^2+p_k+1}$ .

Puisque  $\alpha_k \in I_{(2,p_k)}^2$ , il s'ensuit que  $\alpha_k \geq \frac{p_k}{2p_k^2-p_k+1} \geq \frac{p_k+1}{2p_k^2+p_k+1}$  car  $p_k \geq 2$ .

Si  $n \geq 3$  alors  $\frac{(p_k-1)((p_k+1)(n-1)-1)\alpha_k}{(p_k^3+(n-1)p_k+n-2)\alpha_k+(n-2)p_k^2-p_k-n+2} \leq \frac{p_k+1}{p_k((p_k+1)(n-1)-1)}$  si et seulement si  $\alpha_k \leq \frac{(p_k+1)((n-2)p_k^2-p_k-n+2)}{n(n-2)p_k^4+(n^2-4n+2)p_k^3-(n^2-n-1)p_k^2-(n-1)^2p_k-n+2}$ . Notons que  $n(n-2)p_k^4+(n^2-4n+2)p_k^3-(n^2-n-1)p_k^2-(n-1)^2p_k-n+2 \geq 0$  pour  $p_k \geq 2$  et  $n \geq 3$ . Puisque  $\alpha_k \in I_{(n,p_k)}^2$ , il s'ensuit que

$$\alpha_k < \frac{p_k+1}{p_k((p_k+1)n-1)} \leq \frac{(p_k+1)((n-2)p_k^2-p_k-n+2)}{n(n-2)p_k^4+(n^2-4n+2)p_k^3-(n^2-n-1)p_k^2-(n-1)^2p_k-n+2}$$

si et seulement si  $(n-1)p_k + n - 2 \geq 0$  qui est vrai puisque  $n \geq 3$  et  $p_k \geq 2$ . D'où,

$$\tilde{\alpha}_k \leq \frac{p_k+1}{p_k((p_k+1)(n-1)-1)} \quad (4.19)$$

En utilisant la propriété 4.2 (page 103) si  $n \geq 3$  avec (4.19) ou  $W_1(x) = 1$  si  $n = 2$ , on déduit que  $W_{n-1}(\tilde{\alpha}_k) \geq \frac{p_k}{(p_k+1)(n-1)-1}$ . L'inégalité (4.17) implique que  $\tilde{u}_k(B_k) \geq$

$$\left( \frac{p_k}{(p_k+1)(n-1)-1} \right) \frac{(p_k^3+(n-1)p_k+n-2)\alpha_k+(n-2)p_k^2-p_k-n+2}{(p_k-1)((p_k+1)(n-1)-1)} \geq \frac{p_k(1-p_k\alpha_k)}{(p_k+1)(n-1)-1} = W_n(\alpha_k)$$

si et seulement si  $\alpha_k \geq \frac{p_k^2}{np_k^3-p_k^2+p_k+n-2}$  qui est vrai puisque  $\alpha_k \in I_{(n,p_k)}^2$ .

**(c) :**  $\alpha_k \in I_{(n,p_k)}^3$ . Par construction,  $W_n(\alpha_k) = \frac{p_k(\alpha_k+p_k-1)}{np_k^2-p_k-n+2}$ . D'une part, de (4.13),

$$\tilde{u}_k(B_k) \geq W_{n-1}(\tilde{\alpha}_k) \frac{((n-1)p_k + n - 2)(\alpha_k + p_k - 1)}{np_k^2 - p_k - n + 2} \quad (4.20)$$



D'autre part, de (4.14), on obtient  $\tilde{\alpha}_k \leq \frac{(np_k^2 - p_k - n + 2)\alpha_k}{((n-1)p_k + n - 2)(\alpha_k + p_k - 1)}$ . Montrons que  $\tilde{\alpha}_k \leq \frac{p_k + 1}{p_k((p_k + 1)(n-1) - 1)}$ . L'inégalité  $\frac{(np_k^2 - p_k - n + 2)\alpha_k}{((n-1)p_k + n - 2)(\alpha_k + p_k - 1)} \leq \frac{p_k + 1}{p_k((p_k + 1)(n-1) - 1)}$  est valable si et seulement si  $\alpha_k \leq \frac{p_k^2 - 1}{np_k^3 - p_k^2 - (n-1)p_k - 1}$ . Puisque  $\alpha_k \in I_{(n, p_k)}^3$  alors  $\alpha_k < \frac{p_k^2}{np_k^3 - p_k^2 + p_k + n - 2} \leq \frac{p_k^2 - 1}{np_k^3 - p_k^2 - (n-1)p_k - 1}$  est toujours vrai car la dernière inégalité est équivalente à  $np_k^2 - p_k - n + 2 \geq 0$  qui est valide pour  $n, p_k \geq 2$ . D'où,

$$\tilde{\alpha}_k \leq \frac{p_k + 1}{p_k((p_k + 1)(n-1) - 1)} \quad (4.21)$$

En utilisant la propriété 4.2 (page 103) si  $n \geq 3$  avec (4.21) ou  $W_1(x) = 1$  si  $n = 2$ , on déduit que  $W_{n-1}(\tilde{\alpha}_k) \geq \frac{p_k}{(p_k + 1)(n-1)}$ . L'inégalité (4.20) implique que  $\tilde{u}_k(B_k) \geq \frac{p_k(\alpha_k + p_k - 1)}{np_k^2 - p_k - n + 2} = W_n(\alpha_k)$  car  $\alpha_k \in I_{(n, p_k)}^3$ .

La récurrence est démontrée et le résultat s'ensuit. ■

L'analyse de l'algorithme SEUIL est au plus près (*tight analysis*). Cela signifie qu'il existe des instances critiques (*tight instances*) où l'utilité de l'agent le plus défavorisé de la solution retournée par SEUIL est la même que celle du théorème 4.1. Ces instances sont présentées dans l'annexe C page, 181.

### 4.3 Une légère modification de SEUIL pour deux agents

Pour  $n = 2$  agents, nous proposons un nouvel algorithme appelé MAX-SEUIL qui permet d'améliorer la garantie de l'agent le plus défavorisé comparée à celle donnée par SEUIL. MAX-SEUIL est obtenu en modifiant la valeur du seuil que chaque agent doit atteindre dans l'algorithme 4.1 (page 104). La méthode de sélection de l'agent qui ajoutera ses éléments en premier à la solution est également revue. MAX-SEUIL est décrit dans l'algorithme 4.2.

**Théorème 4.2** *Si  $n = 2$  agents alors MAX-SEUIL produit une base  $B$  avec*

$$u_i(B) \geq \max\{W_2(\alpha_1), W_2(\alpha_2)\}u_i(B_i^*), \quad i = 1, 2$$

**Preuve.** De façon équivalente, nous prouvons que  $\tilde{u}_i(B) \geq \max\{W_2(\alpha_1), W_2(\alpha_2)\}$  pour  $i = 1, 2$  (voir la ligne 4 de MAX-SEUIL). Sans perte de généralité, supposons que  $i = 1$  à la ligne 9 de MAX-SEUIL. Ainsi,

$$|S_1| \leq |S_2| \quad (4.22)$$

---

**Algorithm 4.2 : MAX-SEUIL**


---

**Données :** un matroïde  $\mathcal{M} = (X, \mathcal{F})$ , des utilités  $u_1, u_2$

**Résultat :** Une base-allocation de  $\mathcal{M}$

```

1 pour  $i = 1, 2$  faire
2    $B_i^* \leftarrow \text{GLOUTON}(\mathcal{M}, u_i, \max)$ 
3   pour tout  $e \in X$  faire
4      $\tilde{u}_i(e) \leftarrow \frac{u_i(e)}{u_i(B_i^*)}$ 
5    $\alpha_i \leftarrow \max_{\{e\} \in \mathcal{F}} \tilde{u}_i(e)$ 
6 pour  $i = 1, 2$  faire
7    $S_i \leftarrow \emptyset$ 
8   ajouter de façon gloutonne des éléments de  $B_i^*$  à  $S_i$  par ordre décroissant de  $\tilde{u}_i$  jusqu'à
   ce que  $\tilde{u}_i(S_i) \geq \max\{W_2(\alpha_1), W_2(\alpha_2)\}$ 
9 choisir un agent  $i \in \{1, 2\}$  tel que  $|S_i| < |S_{3-i}|$  ou ( $|S_i| = |S_{3-i}|$  et  $W_2(\alpha_i) \leq W_2(\alpha_{3-i})$ )
10 retourner  $S_i \cup \text{GLOUTON}(\mathcal{M}/S_i, \tilde{u}_{3-i}, \max)$ 

```

---

De plus, nous savons grâce à la ligne 8 de MAX-SEUIL que l'agent 1 atteint la garantie  $\max\{W_2(\alpha_1), W_2(\alpha_2)\}$ . Puisque la contribution de l'agent 1 est  $S_1 \subseteq B$  et puisque  $\tilde{u}_1$  est positive, alors  $\tilde{u}_1(B) \geq \tilde{u}_1(S_1) \geq \max\{W_2(\alpha_1), W_2(\alpha_2)\}$ .

Soit  $\tilde{S}_2$  la contribution de l'agent 2 produite par  $\text{GLOUTON}(\mathcal{M}/S_1, \tilde{u}_2, \max)$  à la ligne 10 de MAX-SEUIL. Considérons les deux scénarios qui peuvent se produire à la ligne 9.

- Si  $|S_1| = |S_2|$  alors on déduit de la ligne 9 que  $W_2(\alpha_1) \leq W_2(\alpha_2)$ . Du point de vue de l'agent 2, appliquer MAX-SEUIL  $(\mathcal{M}, u_1, u_2)$  est équivalent à appliquer SEUIL  $(\{1, 2\}, \mathcal{M}, (u_2, u_2), \emptyset)$ . Comme nous l'avons montré dans le théorème 4.1 (page 106), la contribution de l'agent 2 est  $\tilde{u}_2(\tilde{S}_2) \geq W_2(\alpha_2) = \max\{W_2(\alpha_1), W_2(\alpha_2)\}$ .
- Si  $|S_1| < |S_2|$  alors en utilisant les mêmes arguments que dans la preuve du théorème 4.1 (page 106), on obtient

$$\tilde{u}_2(\tilde{S}_2) \geq 1 - \tilde{u}_2(B_2^{*1}) \quad (4.23)$$

où  $B_2^{*1}$  contient les  $|S_1|$  premiers éléments de poids maximum de  $B_2^*$  (voir la ligne 2 de MAX-SEUIL). Alors,  $B_2^{*1} \subseteq S_2$  pour les raisons suivantes :  $S_2 \subseteq B_2^*$ ,  $|B_2^{*1}| = |S_1| < |S_2|$ ,  $S_2$  et  $B_2^*$  sont construits en ajoutant les éléments par ordre décroissant de  $\tilde{u}_2$ . On conclut que  $S_2$  est minimal pour l'inclusion car ses éléments sont ajoutés graduellement, donc  $\tilde{u}_2(B_2^{*1}) < \max\{W_2(\alpha_1), W_2(\alpha_2)\}$ . Finalement, (4.23) devient

$$\tilde{u}_2(\tilde{S}_2) \geq 1 - \max\{W_2(\alpha_1), W_2(\alpha_2)\} \geq \max\{W_2(\alpha_1), W_2(\alpha_2)\}$$

où la dernière inégalité est valide si et seulement si  $\max\{W_2(\alpha_1), W_2(\alpha_2)\} \leq \frac{1}{2}$  qui est vrai grâce à la propriété 4.1 (page 103).

Dans les deux cas,  $\tilde{u}_2(\tilde{S}_2) \geq \max\{W_2(\alpha_1), W_2(\alpha_2)\}$ . En utilisant la positivité de  $\tilde{u}_2$ , on conclut que  $\tilde{u}_2(B) \geq \tilde{u}_2(\tilde{S}_2) \geq \max\{W_2(\alpha_1), W_2(\alpha_2)\}$ . ■

## 4.4 Borne supérieure sur l'utilité de l'agent le moins heureux

Les résultats positifs des sections 4.2 et 4.3 fournissent des bornes inférieures sur l'utilité de chaque agent et sont liés au vecteur  $(\alpha_i)_{i \in [n]}$ . En vue du nombre important de valeurs que peut prendre le vecteur  $(\alpha_i)_{i \in [n]}$ , nous fournissons une borne supérieure à l'agent le plus défavorisé et nous exprimons cette borne comme une fonction de  $\alpha = \max_{i \in N} \alpha_i$  et  $n$ .

Nous présentons dans cette section des instances qui fournissent cette borne supérieure, notée  $\overline{W}_n(\alpha)$ , qui représente l'utilité maximum que l'agent le plus défavorisé puisse avoir. La borne supérieure est calculée de la façon suivante :

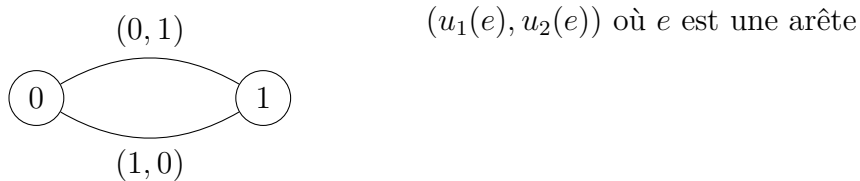
$$\overline{W}_n(\alpha) = \max_{B \in \mathcal{B}} \min_{i \in N} u_i(B) \quad (4.24)$$

où  $\mathcal{B}$  est l'ensemble des bases du matroïde.

Toutes les instances fournies dans cette section sont des instances du matroïde graphique où une base correspond à un arbre couvrant. On dira qu'une instance est bien définie si pour toute arête  $e$  du graphe correspondant,  $\alpha \geq u_i(e) \geq 0$  pour tout  $i \in N$ . Nous supposons que toutes les instances sont normalisées.

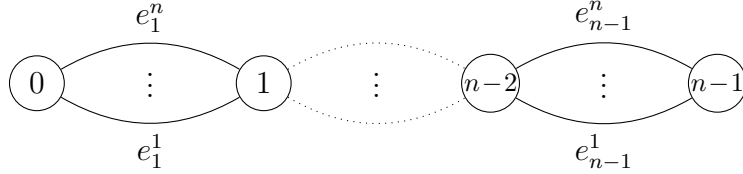
Nous obtenons les résultats suivants selon les valeurs que peut prendre  $\alpha \in [0, 1]$ .

1. Si  $\alpha \in I_{(n,0)} = \left[\frac{1}{n-1}, 1\right]$  alors  $\overline{W}_n(\alpha) = 0$ , c'est-à-dire qu'il n'y a pas de garantie, comme on peut l'observer sur les instances suivantes du matroïde graphique selon que  $n = 2$  ou  $n \geq 3$ .
  - (a) Si  $n = 2$ , alors considérons le multigraphe ci-dessous :



Puisque le graphe est connexe, alors il suffit de choisir l'une des deux arêtes pour obtenir un arbre couvrant. L'utilité de l'agent le plus défavorisé est  $\min\{0, 1\} = 0$ , donc  $\overline{W}_2(\alpha) = \overline{W}_2(1) = 0 = W_2(\alpha)$ .

(b) Si  $n \geq 3$ , alors considérons l'instance suivante :



Nous avons ici un multigraphe  $G = (V, E)$  tel que  $V = \{0, 1, \dots, n-1\}$  est l'ensemble des sommets et  $E = \bigcup_{k=1}^{n-1} \{e_k^1, \dots, e_k^n\}$  est l'ensemble des arêtes. Il y a  $n$  arêtes entre chaque paire de nœuds consécutifs. Pour tout  $i, j \in \{1, \dots, n\}$  :  $u_i(e_1^j) = \alpha$  si  $i = j$  et  $u_i(e_1^j) = 0$  sinon ; et pour tout  $k \in \{2, \dots, n-1\}$ ,  $u_i(e_k^j) = \frac{1-\alpha}{n-2}$  si  $i = j$  et  $u_i(e_k^j) = 0$  sinon.

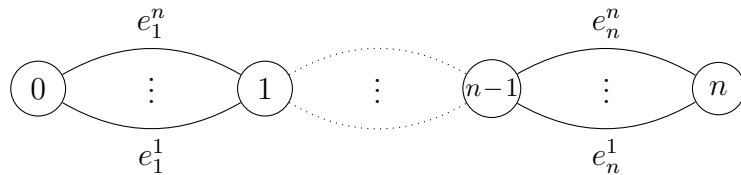
L'instance est bien définie pour les raisons suivantes :

- $\alpha \geq \frac{1}{n-1} > 0$  car  $n \geq 3$  ;
- $\frac{1-\alpha}{n-2} \geq 0$  car  $\alpha \leq 1$  et  $n \geq 3$  ;
- $\alpha \geq \frac{1-\alpha}{n-2}$  car  $\alpha \geq \frac{1}{n-1}$ .

Pour tout  $i \in \{1, \dots, n\}$ , soit  $B_i^* = \{e_1^i, \dots, e_{n-1}^i\}$  l'arbre couvrant optimal pour l'agent  $i$  tel que  $u_i(B_i^*) = \sum_{j=1}^{n-1} u_i(e_j^i) = \alpha + (n-2)\frac{1-\alpha}{n-2} = 1$ . Alors l'instance est normalisée.

Pour toute arête  $e$ , il existe au plus un  $i \in \{1, \dots, n\}$  tel que  $u_i(e) > 0$ , sinon  $u_j(e) = 0$  pour tout  $j \in \{1, \dots, n\} \setminus \{i\}$ . Un arbre couvrant contient  $n-1$  arêtes. Alors pour tout arbre couvrant  $B$ , il existe  $k \in \{1, \dots, n\}$  tel que  $u_k(B) = 0$ . D'où,  $\overline{W}_n(\alpha) = 0 = W_n(\alpha)$ .

2. Si  $\alpha \in I_{(n,1)}^1 = \left[ \frac{2}{2n-1}, \frac{1}{n-1} \right[$  alors  $\overline{W}_n(\alpha) = 1 - (n-1)\alpha$ , comme on peut l'observer sur l'instance suivante :



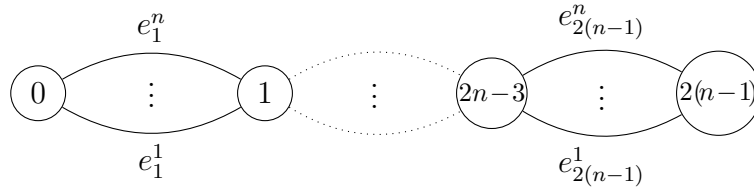
Pour tout  $i, j \in \{1, \dots, n\}$  et pour tout  $k \in \{1, \dots, n-1\}$ ,  $u_i(e_k^j) = \alpha$  si  $i = j$  et  $u_i(e_k^j) = 0$  sinon. De plus,  $u_i(e_n^j) = 1 - (n-1)\alpha$  si  $i = j$  et  $u_i(e_n^j) = 0$  sinon. L'instance est bien définie pour les raisons suivantes :

- $\alpha \geq \frac{2}{2n-1} > 0$  car  $n \geq 2$ ;
- $1 - (n-1)\alpha > 0$  car  $\alpha < \frac{1}{n-1}$ ;
- $\alpha \geq 1 - (n-1)\alpha$  si et seulement si  $\alpha \geq 1/n$  et par hypothèse  $\alpha \geq \frac{2}{2n-1} \geq \frac{1}{n}$ .

Soit  $B_i^* = \{e_1^i, \dots, e_n^i\}$  un arbre couvrant optimal pour chaque agent  $i \in \{1, \dots, n\}$ . Alors  $u_i(B_i^*) = \sum_{j=1}^n u_i(e_j^i) = (n-1)\alpha + 1 - (n-1)\alpha = 1$ , donc l'instance est normalisée.

Chaque arête de l'instance a une utilité strictement positive pour un unique agent. Puisqu'un arbre couvrant contient  $n$  arêtes, il faut choisir  $n$  arêtes  $\{(i, i+1), i = 1, \dots, n-1\}$ . L'un des arbres couvrants qui maximisent l'utilité de l'agent le plus défavorisé est  $T = \{e_i^i, i = 1, \dots, n\}$  tel que  $u_i(T) = \alpha$  pour tout  $i \in \{1, \dots, n-1\}$  et  $u_n(T) = 1 - (n-1)\alpha$ . D'où  $\overline{W}_n(\alpha) = \min\{\alpha, 1 - (n-1)\alpha\} = 1 - (n-1)\alpha$  si et seulement si  $\alpha \geq 1/n$ . Par hypothèse,  $\alpha \geq \frac{2}{2n-1} \geq \frac{1}{n}$  est toujours vrai, donc  $\overline{W}_n(\alpha) = 1 - (n-1)\alpha = W_n(\alpha)$ .

3. Si  $\alpha \in I_{(n,1)}^2 = \left[ \frac{1}{2(n-1)}, \frac{2}{2n-1} \right[$  alors  $\overline{W}_n(\alpha) = \frac{1-\alpha}{2n-3}$ , comme on peut l'observer sur l'instance suivante :



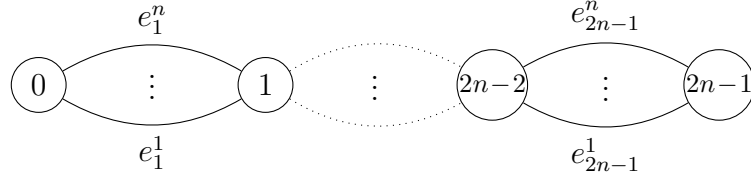
Pour tout  $i, j \in \{1, \dots, n\}$  :  $u_i(e_1^j) = \alpha$  si  $i = j$  et  $u_i(e_1^j) = 0$  sinon. Pour tout  $k \in \{2, \dots, 2(n-1)\}$ ,  $u_i(e_k^j) = \frac{1-\alpha}{2n-3}$  si  $i = j$  et  $u_i(e_k^j) = 0$  sinon. L'instance est bien définie pour les raisons suivantes :

- $\alpha \geq \frac{1}{2(n-1)} > 0$  car  $n \geq 2$ ;
- $\frac{1-\alpha}{2n-3} > 0$  car  $\alpha < \frac{2}{2n-1} < 1$  et  $n \geq 2$ ;
- $\alpha \geq \frac{1-\alpha}{2n-3}$  car  $\alpha \geq \frac{2}{2(n-1)}$ .

Soit  $B_i^* = \{e_1^i, \dots, e_{2(n-1)}^i\}$  un arbre couvrant optimal pour l'agent  $i$  pour tout  $i \in \{1, \dots, n\}$ . Alors  $u_i(B_i^*) = \sum_{j=1}^{2(n-1)} u_i(e_j^i) = \alpha + (2n-3)\frac{1-\alpha}{2n-3} = 1$ , donc l'instance est normalisée.

L'arbre couvrant  $T = \{e_1^1, e_2^2\} \cup \{e_{2(i-1)-1}^i, e_{2(i-1)}^i : i = 3, \dots, n\}$  maximise l'utilité de l'agent le plus défavorisé tel que  $u_1(T) = \alpha \geq \frac{1-\alpha}{2n-3}$  car  $\alpha \geq \frac{1}{2(n-1)}$ ,  $u_2(T) = \frac{1-\alpha}{2n-3}$  et  $u_i(T) = 2\frac{1-\alpha}{2n-3} > \frac{1-\alpha}{2n-3}$  pour  $i = 3, \dots, n$ . D'où,  $\overline{W}_n(\alpha) = \min\left\{\alpha, \frac{1-\alpha}{2n-3}, 2\frac{1-\alpha}{2n-3}\right\} = \frac{1-\alpha}{2n-3} = W_n(\alpha)$ .

4. Si  $\alpha \in I_{(n,1)}^3 = \left[ \frac{1}{2n-1}, \frac{1}{2(n-1)} \right]$  alors  $\overline{W}_n(\alpha) = \alpha$ , comme on peut l'observer sur l'instance suivante :



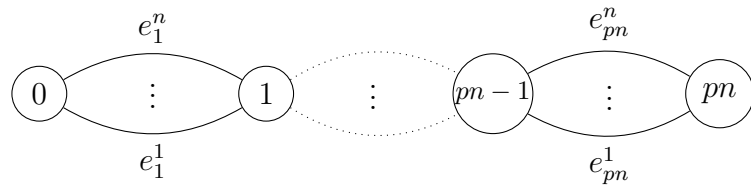
Pour tout  $i, j \in \{1, \dots, n\} : u_i(e_1^j) = \alpha$  si  $i = j$  et  $u_i(e_1^j) = 0$  sinon. Pour tout  $k \in \{2, \dots, 2n-1\}$ ,  $u_i(e_k^j) = \frac{1-\alpha}{2(n-1)}$  si  $i = j$  et  $u_i(e_k^j) = 0$  sinon. L'instance est bien définie pour les raisons suivantes :

- $\alpha \geq \frac{1}{2n-1} > 0$  car  $n \geq 2$  ;
- $\frac{1-\alpha}{2(n-1)} > 0$  car  $\alpha < \frac{1}{2(n-1)} < 1$  et  $n \geq 2$  ;
- $\alpha \geq \frac{1-\alpha}{2(n-1)}$  car  $\alpha \geq \frac{1}{2n-1}$ .

Soit  $B_i^* = \{e_1^i, \dots, e_{2n-1}^i\}$  un arbre couvrant optimal pour l'agent  $i$  pour tout  $i \in \{1, \dots, n\}$  tel que  $u_i(B_i^*) = \sum_{j=1}^{2n-1} u_i(e_j^i) = \alpha + 2(n-1)\frac{1-\alpha}{2(n-1)} = 1$ . L'instance est normalisée.

L'arbre couvrant  $T = \{e_1^1\} \cup \{e_{2(i-1)}^i, e_{2i-1}^i : i = 2, \dots, n\}$  maximise l'utilité de l'agent le plus défavorisé tel que  $u_1(T) = \alpha$  et  $u_i(T) = 2\frac{1-\alpha}{2(n-1)}$  pour  $i = 2, \dots, n$ . Alors  $\overline{W}_n(\alpha) = \min \left\{ \alpha, \frac{1-\alpha}{n-1} \right\} = \alpha$  si et seulement si  $\alpha \geq 1/n$  et par hypothèse  $\alpha < \frac{1}{2(n-1)} \leq \frac{1}{n}$  car  $n \geq 2$ . D'où  $\overline{W}_n(\alpha) = \alpha = W_n(\alpha)$ .

5. Si  $\alpha \in \left[ \frac{p+1}{np^2+(n-1)p+1}, \frac{1}{pn-1} \right]$  pour un entier  $p \geq 2$  alors  $\overline{W}_n(\alpha) = 1 - p(n-1)\alpha$ , comme on peut l'observer sur l'instance suivante :



Pour tout  $i, j \in \{1, \dots, n\} : u_i(e_k^j) = \alpha$  si  $i = j$  et  $u_i(e_k^j) = 0$  sinon. De plus,  $u_i(e_{pn}^j) = 1 - (pn-1)\alpha$  si  $i = j$  et  $u_i(e_{pn}^j) = 0$  sinon. L'instance est bien définie pour les raisons suivantes :

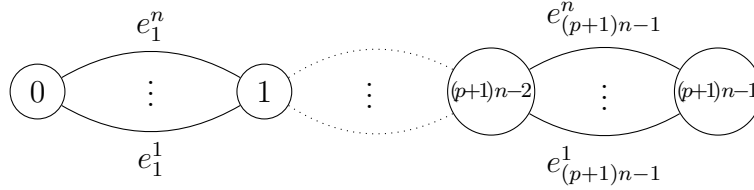
- $\alpha \geq \frac{p+1}{np^2+(n-1)p+1} > 0$  car  $p, n \geq 2$  ;
- $1 - (pn-1)\alpha > 0$  car  $\alpha < \frac{1}{pn-1}$  ;
- $\alpha \geq 1 - (pn-1)\alpha$  si et seulement si  $\alpha \geq \frac{1}{pn}$  et par hypothèse  $\alpha \geq \frac{p+1}{np^2+(n-1)p+1} \geq \frac{1}{pn}$  car  $p \geq 2$ .

Soit  $B_i^* = \{e_1^i, \dots, e_{pn}^i\}$  un arbre couvrant optimal pour l'agent  $i \in \{1, \dots, n\}$  tel que  $u_i(B_i^*) = \sum_{j=1}^{pn} u_i(e_j^i) = (pn-1)\alpha + 1 - (pn-1)\alpha = 1$ . Alors l'instance est normalisée.

L'arbre couvrant  $T = \cup_{i=1}^n \{e_{(i-1)p+1}^i, \dots, e_{ip}^i\}$  maximise l'utilité de l'agent le plus défavorisé tel que  $u_i(T) = p\alpha$  pour  $i = 1, \dots, n-1$  et  $u_n(T) = (p-1)\alpha + 1 - (pn-1)\alpha = 1 - p(n-1)\alpha$ . Alors  $\overline{W}_n(\alpha) = \min\{p\alpha, 1 - p(n-1)\alpha\} = 1 - p(n-1)\alpha$  si et seulement si  $\alpha \geq \frac{1}{pn}$  et par hypothèse  $\alpha \geq \frac{p+1}{np^2+(n-1)p+1} \geq \frac{1}{pn}$  car  $p \geq 2$ .

Puisque  $\frac{p+1}{np^2+(n-1)p+1} < \frac{p+1}{p((p+1)n-1)}$ , il s'ensuit que  $I_{(n,p)}^1 \subset \left[ \frac{p+1}{np^2+(n-1)p+1}, \frac{1}{pn-1} \right]$ , donc si  $\alpha \in I_{(n,p)}^1$  alors  $\overline{W}_n(\alpha) = 1 - p(n-1)\alpha = W_n(\alpha)$ .

6. Si  $\alpha \in \left[ \frac{1}{(p+1)n-1}, \frac{p+1}{np^2+(n-1)p+1} \right]$  pour un entier  $p \geq 2$  alors  $\overline{W}_n(\alpha) = \alpha + \frac{(p-1)(1-\alpha)}{(p+1)n-2}$ , comme on peut l'observer sur l'instance suivante :



Pour tout  $i, j \in \{1, \dots, n\} : u_i(e_j^i) = \alpha$  si  $i = j$  et  $u_i(e_j^i) = 0$  sinon. Pour tout  $k \in \{2, \dots, (p+1)n-1\}$ ,  $u_i(e_k^j) = \frac{1-\alpha}{(p+1)n-2}$  si  $i = j$  et  $u_i(e_k^j) = 0$  sinon. L'instance est bien définie pour les raisons suivantes :

- $\alpha \geq \frac{1}{(p+1)n-1} > 0$  car  $p, n \geq 2$ ;
- $\frac{1-\alpha}{(p+1)n-2} > 0$  car  $\alpha < \frac{p+1}{np^2+(n-1)p+1} < 1$  et  $p, n \geq 2$ ;
- $\alpha \geq \frac{1-\alpha}{(p+1)n-2}$  car  $\alpha \geq \frac{1}{(p+1)n-1}$ .

Soit  $B_i^* = \{e_1^i, \dots, e_{(p+1)n-1}^i\}$  un arbre couvrant optimal pour l'agent  $i$  pour tout  $i \in \{1, \dots, n\}$  tel que  $u_i(B_i^*) = \sum_{j=1}^{(p+1)n-1} u_i(e_j^i) = \alpha + ((p+1)n-2) \frac{1-\alpha}{(p+1)n-2} = 1$ . Alors l'instance est normalisée.

L'arbre couvrant  $T = \{e_1^1, \dots, e_p^1\} \cup_{i=1}^{n-1} \{e_{(p+1)i}^{i+1}, \dots, e_{(p+1)(i+p)}^{i+1}\}$  maximise l'utilité de l'agent le plus défavorisé tel que  $u_1(T) = \alpha + \frac{(p-1)(1-\alpha)}{(p+1)n-2}$  et  $u_i(T) = \frac{(p+1)(1-\alpha)}{(p+1)n-2}$  pour  $i = 2, \dots, n$ . Alors  $\overline{W}_n(\alpha) = \min \left\{ \alpha + \frac{(p-1)(1-\alpha)}{(p+1)n-2}, \frac{(p+1)(1-\alpha)}{(p+1)n-2} \right\} = \alpha + \frac{(p-1)(1-\alpha)}{(p+1)n-2}$  si et seulement si  $\alpha \leq \frac{2}{(p+1)n}$  et par hypothèse  $\alpha < \frac{p+1}{np^2+(n-1)p+1} \leq \frac{2}{(p+1)n}$  car  $p(np-2) + 2 - n \geq 0$  puisque  $p, n \geq 2$ .

### 4.4.1 Conclusion sur la valeur de $t_n$

Nous avons vu dans l'introduction (page 98) que la valeur  $t_n \in [0, 1]$  représente l'utilité maximum que chaque agent puisse avoir. Dans cette section, nous concluons sur la valeur de  $t_n$  en tenant compte des bornes inférieures fournies par les algorithmes SEUIL et MAX-SEUIL, et de la borne supérieure  $\overline{W}_n(\alpha)$  présentée dans la Section 4.4 où  $\alpha = \max_{i \in [n]} \alpha_i$ .

#### La valeur de $t_2$

Lorsque  $n = 2$ , nous avons vu dans la section 4.3 que  $\max\{W_2(\alpha_1), W_2(\alpha_2)\}$  est une borne inférieure sur l'utilité de l'agent le plus défavorisé. Cette garantie est fournie par MAX-SEUIL. Par ailleurs,  $\overline{W}_2(\alpha)$  est une borne supérieure sur l'utilité de l'agent le plus défavorisé. Afin de comparer ces deux bornes et puisque  $W_n$  est non-monotone, nous supposons que  $\alpha_1 = \alpha_2 = \alpha$ , donc  $\max\{W_2(\alpha_1), W_2(\alpha_2)\} = W_2(\alpha)$ . La différence entre les bornes inférieure et supérieure est illustrée sur la figure 4.4.

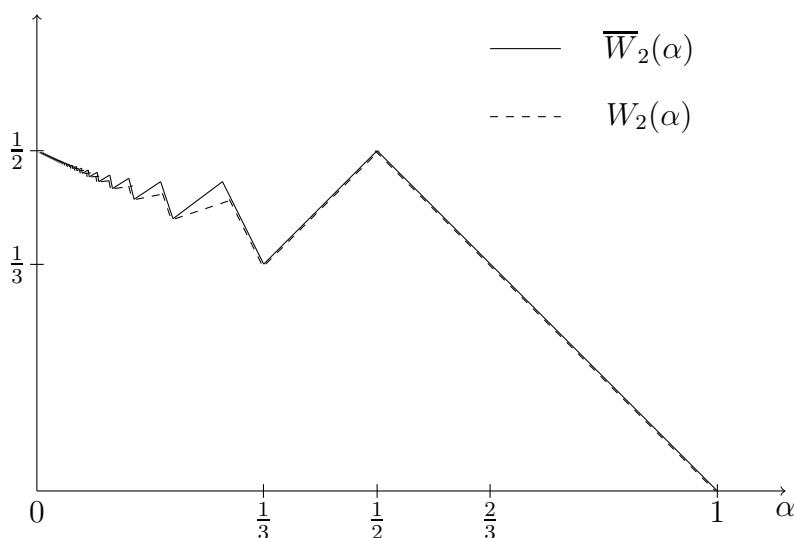


FIGURE 4.4 – Les fonctions  $W_2(\alpha)$  et  $\overline{W}_2(\alpha)$ .

Si  $\alpha = \alpha_1 = \alpha_2$  alors  $t_2 = \overline{W}_2(\alpha) = W_2(\alpha)$  lorsque  $\alpha \in I_{(2,0)} \cup I_{(2,1)} \cup I_{(2,p)}^1 \cup I_{(2,p)}^2$  pour tout entier  $p \geq 2$ . Sinon,  $W_2(\alpha) \leq t_2 \leq \overline{W}_2(\alpha)$  pour  $\alpha \in I_{(2,p)}^3$  où  $p \geq 2$  est un entier.



### La valeur de $t_n$ pour $n \geq 3$ agents

Lorsque  $n \geq 3$ , nous avons vu dans la section 4.2 que  $W_n(\alpha_i)$  est une borne inférieure sur l'utilité de l'agent  $i$ , produite par SEUIL. Puisque  $W_n$  est non-monotone, nous ne pouvons pas conclure que la garantie  $W_n(\alpha)$  est aussi valide pour tous les agents car, même si  $\alpha = \max_{i \in [n]} \alpha_i$ , nous ne pouvons rien dire sur  $W_n(\alpha)$  comparé à  $W_n(\alpha_i)$ . Ainsi, dans le but de pouvoir comparer la garantie  $W_n(\alpha_i)$  à la borne supérieure  $\overline{W}_n(\alpha)$  pour  $n \geq 3$ , nous supposons que  $\alpha = \alpha_i$  pour tout  $i \in N$ .

Le tableau 4.3 contient une synthèse des bornes inférieure et supérieure sur l'utilité de l'agent le plus défavorisé en fonction de  $\alpha$  lorsque  $\alpha = \alpha_i, \forall i \in N$ .

Comme on peut l'observer dans le tableau 4.3, si  $\alpha = \alpha_i$  pour tout  $i \in N$  alors  $t_n = \overline{W}_n(\alpha) = W_n(\alpha)$  sur  $I_{(n,0)} \cup I_{(n,1)} \cup I_{(n,p)}^1$  pour tout entier  $p \geq 2$ . Sinon,  $W_n(\alpha) \leq t_n \leq \overline{W}_n(\alpha)$  pour  $\alpha \in I_{(n,p)}^2 \cup I_{(n,p)}^3$  pour tout entier  $p \geq 2$ . Demko et Hill (1988); Markakis et Psomas (2011) ont montré que la garantie  $V_n(\alpha)$  est maximum uniquement sur  $I_{(n,p)}^1$  pour tous les entiers  $n \geq 2$  et  $p \geq 1$ .

## Conclusion et perspectives

L'algorithme SEUIL fournit une solution qui est une base  $B$  d'un matroïde avec une utilité relative d'au moins  $W_n(\alpha_i)$  pour chaque agent  $i \in \{1, \dots, n\}$  et pour n'importe quel nombre d'agents  $n \geq 1$ . Cette garantie satisfait, pour des entiers  $n \geq 2$  et  $p \geq 1$ ,  $W_n(\alpha_i) = V_n(\alpha_i)$  pour tout  $\alpha_i \in I_{(n,p)}^1$  et  $W_n(\alpha_i) > V_n(\alpha_i)$  pour tout  $\alpha_i \in I_{(n,p)}^2 \cup I_{(n,p)}^3$  où  $V_n(\alpha_i)$  est la garantie donnée par Markakis et Psomas (2011) sur l'utilité de l'agent  $i$  pour le problème classique d'allocation de biens indivisibles avec des utilités additives. La garantie  $(W_n(\alpha_i))_{i \in [n]}$  est valide pour le problème de la base d'un matroïde multi-agents (ou multi-objectif) et qui généralise le problème classique d'allocation de biens indivisibles. Notons que cette garantie représente une approximation du point idéal du problème de la base d'un matroïde de poids maximum à  $n$  objectifs.

Dans ce chapitre, nous avons fourni des bornes inférieure et supérieure sur la valeur  $t_n$ , représentant l'utilité que chaque agent peut avoir. Cependant, ces bornes ne coïncident pas pour toutes les valeurs de  $\alpha_i \in [0, 1]$ . Une première direction consiste à réduire l'écart entre les bornes inférieure et supérieure proposées.

Contrairement à  $V_n$ ,  $W_n$  est non-monotone. Ce fait est dû à l'utilisation d'éléments discrets/indivisibles. Il n'est donc pas évident de garantir une utilité relative de  $W_n(\alpha)$  à chaque agent. Cependant, pour  $n = 2$  agents, l'application de MAX-SEUIL (voir la section 4.3) permet de fournir une garantie d'au moins  $\max\{W_2(\alpha_1), W_2(\alpha_2)\}$  à chaque agent  $i \in N = \{1, 2\}$ . Un défi intéressant serait de savoir si la borne  $\max_{i \in N} W_n(\alpha_i)$  peut être atteinte par chaque agent  $i \in N$  lorsque  $n \geq 3$ .

	$p \geq 2$			$p = 1$			$p = 0$
Intervalle de $\alpha$	$I_{(n,p)}^3$	$I_{(n,p)}^2$	$I_{(n,p)}^1$	$I_{(n,p)}^3$	$I_{(n,p)}^2$	$I_{(n,p)}^1$	$I_{(n,p)}$
Borne inférieure $W_n(\alpha)$	$\frac{p(\alpha+p-1)}{np^2-p-n+2}$	$\frac{p(1-p\alpha)}{(p+1)(n-1)-1}$	$1 - p(n-1)\alpha$	$\alpha$	$\frac{1-\alpha}{2n-3}$	$1 - (n-1)\alpha$	0
	$p \geq 2$			$p = 1$			$p = 0$
Intervalle de $\alpha$	$\left[ \frac{1}{(p+1)n-1}, \frac{p+1}{np^2+(n-1)p+1} \right]$	$\left[ \frac{p+1}{np^2+(n-1)p+1}, \frac{1}{pn-1} \right]$		$I_{(n,p)}^3$	$I_{(n,p)}^2$	$I_{(n,p)}^1$	$I_{(n,p)}$
Borne supérieure $\overline{W}_n(\alpha)$	$\alpha + \frac{(p-1)(1-\alpha)}{(p+1)n-2}$	$1 - p(n-1)\alpha$		$\alpha$	$\frac{1-\alpha}{2n-3}$	$1 - (n-1)\alpha$	0

Tableau 4.3 – Bornes supérieure et inférieure sur l'utilité de l'agent le plus défavorisé.



# Garantie dans le pire cas : une approche décentralisée

Le présent chapitre traite du problème de la base d'un matroïde multi-agents avec une garantie dans le pire cas sur l'utilité de chacun. Ce problème a également été étudié dans le chapitre 4. Les agents ont des utilités pour les éléments du matroïde et ces utilités sont toutes additives. Nous supposons dans ce chapitre que les utilités sont privées, hypothèse souvent faite pour le partage équitable de biens divisibles. Puisque l'utilisation d'algorithmes centralisés n'est pas appropriée dans ce contexte, nous avons recours à des protocoles, comme dans les problèmes de partage de gâteaux. Notre contribution est un algorithme décentralisé où les agents peuvent interagir et construire ensemble une base d'un matroïde. S'il y a au plus 9 agents, nous montrons comment chaque agent peut faire en sorte que son utilité dans le pire cas soit au moins  $V_n(\alpha_i)$  qui est la garantie de Markakis et Psomas (2011) pour le problème classique d'allocation de biens indivisibles, basée sur la garantie de Hill (1987); Demko et Hill (1988).

Ce chapitre a été soumis à une revue internationale. Une version antérieure apparaît dans les actes de la conférence internationale "The 9th Conference on Web and INternet Economics (WINE 2013)" Gourvès *et al.* (2013b).

## Sommaire

---

<b>Introduction</b> . . . . .	<b>98</b>
<b>4.1 Le modèle</b> . . . . .	<b>100</b>
4.1.1 Les fonctions $W_n$ . . . . .	101
<b>4.2 L'algorithme SEUIL</b> . . . . .	<b>103</b>
<b>4.3 Une légère modification de SEUIL pour deux agents</b> . .	<b>112</b>
<b>4.4 Borne supérieure sur l'utilité de l'agent le moins heureux</b> . . . . .	<b>114</b>
4.4.1 Conclusion sur la valeur de $t_n$ . . . . .	119
<b>Conclusion et perspectives</b> . . . . .	<b>120</b>

---

## Introduction

Dans les problèmes d'allocation de ressources, la distinction entre biens divisibles et indivisibles est typiquement faite. Bien qu'un grand nombre de publications soit dévoué au cas divisible (Steinhaus, 1948; Brams et Taylor, 1995; Young, 1995; Brams et Taylor, 1996; Moulin, 2004; Procaccia, 2013), la communauté informatique porte son attention sur l'allocation de biens indivisibles, en particulier lorsqu'il n'y a pas de compensation monétaire (Lipton *et al.*, 2004; Golovin, 2005; Bezáková et Dani, 2005; Bouveret *et al.*, 2005; Bansal et Sviridenko, 2006; Khot et Ponnuswami, 2007; Asadpour et Saberi, 2010; Markakis et Psomas, 2011; Kalinowski *et al.*, 2013b).

Une autre information importante porte sur les utilités des agents : sont-elles publiques ou privées? En effet, les agents peuvent refuser de communiquer leurs utilités respectives à des tiers si ces dernières sont confidentielles. Une telle approche peut également être intéressante si l'on souhaite limiter les tentatives de manipulation du processus de partage. Dans le cas d'utilités publiques, un algorithme centralisé peut déterminer l'allocation. Dans le cas d'utilités privées, les agents peuvent prendre part dans la détermination de la solution finale via un *algorithme décentralisé* appelé aussi *protocole* (voir la section 2.4.1, page 67).

Comme nous l'avons mentionné dans le chapitre précédent, Demko et Hill (1988) s'intéressent au problème classique d'allocation de biens indivisibles avec des utilités additives. Ils montrent l'existence d'une allocation avec une garantie sur l'utilité de l'agent le plus défavorisé. Cette garantie est  $V_n(\alpha)$  où  $n$  est le nombre d'agents et  $\alpha$  est défini par l'utilité maximum d'un bien, parmi les utilités de tous les agents. Markakis et Psomas (2011) revisitent les travaux de Demko et Hill (1988) avec une approche constructive. Ils proposent un algorithme polynomial appelé ALLOCATE (algorithme 2.1, page 65) qui retourne une allocation où l'utilité de l'agent  $i$  est d'au moins  $V_n(\alpha_i) \geq V_n(\alpha)$  où  $\alpha_i$  est l'utilité maximum de l'agent  $i$  pour un bien.

Les chapitres 3 et 4 ont permis de mettre en évidence le fait que le problème classique d'allocation de biens indivisibles peut être étendu à la détermination d'une base d'un matroïde. Cela a permis d'obtenir une nouvelle garantie  $W_n(\alpha_i)$  pour chaque agent  $i$  où  $\alpha_i$  est l'utilité maximum de l'agent  $i$  pour un élément du matroïde. Cette garantie est obtenue avec l'algorithme polynomial déterministe SEUIL (voir l'algorithme 4.1, page 104) qui est une extension de ALLOCATE.

Les algorithmes ALLOCATE et SEUIL sont centralisés et utilisent des utilités connues publiquement car ils prennent en entrée les utilités des agents. Le problème qui consiste à allouer  $m$  biens indivisibles sous des utilités *privées* et *additives* a été étudié par Bezáková et Dani (2005) pour deux agents. Elles proposent une version aléatoire du paradigme CUT AND CHOOSE (voir la section 2.4.1, page 67) et prouvent qu'en espérance, l'utilité de l'agent le plus défavorisé est au moins la

moitié de l'allocation optimale selon le critère égalitariste. Bezáková et Dani ont recours à l'aléatoire car, comme mentionné plus tôt, l'utilité dans le pire cas d'un agent est 0 si  $\alpha$  n'est pas pris en compte ( $V_n(\alpha) = 0$  lorsque  $\alpha \geq \frac{1}{n-1}$ ). Le problème qui consiste à étendre le protocole à un grand nombre d'agents est inscrit comme perspective par Bezáková et Dani (2005).

Dans ce chapitre, nous étudions le cas d'utilités *privées* dans le contexte généralisé des matroïdes. Ainsi, nous empruntons quelques idées des protocoles de partage de gâteaux. Nous proposons un algorithme déterministe décentralisé (dit aussi protocole de partage) pour trouver une base d'un matroïde telle que l'utilité de chaque agent  $i$  est au moins  $V_n(\alpha_i)$ .

À notre connaissance, il n'y a pas de travaux antérieurs sur des protocoles pour les matroïdes. En outre, les protocoles existants pour le problème classique de partage équitable de biens indivisibles ne sont pas directement comparables à nos travaux. La plupart d'entre eux permettent la compensation monétaire ou d'autres types d'hypothèses comme la linéarité forte (voir la section 2.4.1, page 67). Le résultat qui est le plus proche de notre travail est la version aléatoire de CUT AND CHOOSE pour deux agents de Bezáková et Dani (2005).

Ce chapitre est organisé comme suit. Une présentation du modèle est donnée dans la section 5.1. Notre première contribution consiste en deux protocoles pour 2 et 3 agents présentés respectivement dans les sections 5.2 et 5.3. Ils introduisent un protocole général présenté dans la section 5.4 qui est valide pour au plus 9 agents. Nous concluons ce chapitre avec une discussion et quelques perspectives.

## 5.1 Le modèle

Notre problème prend en entrée un matroïde  $\mathcal{M} = (X, \mathcal{F})$  et un ensemble d'agents  $N = \{1, \dots, n\}$ . Le résultat est une unique base  $B$  de  $\mathcal{F}$  qui est partagée par les agents. Chaque agent  $i \in N$  a une utilité positive  $u_i(e)$  pour tout élément  $e \in X$ . Les fonctions d'utilité des agents sont additives. Ces fonctions d'utilité sont également privées, donc la base résultante n'est pas déterminée par un algorithme centralisé. Elle est plutôt construite pas les agents à travers un protocole.

La base  $B$  peut être décomposée en  $n$  sous-ensembles disjoints  $B_1, \dots, B_n$  tels que pour tout  $i \in N$ ,  $B_i$  est la *contribution* de l'agent  $i$ . Une analyse dans le pire cas devant être faite, nous pouvons supposer que l'utilité de n'importe quel agent pour  $B$  est réduite à son utilité pour sa contribution. En effet,  $u_i(B) = u_i(B_i) + u_i(B \setminus B_i) \geq u_i(B_i)$ .

Outre son exactitude, le protocole doit être *équitable*, c'est-à-dire qu'il offre une garantie sur l'utilité de chaque agent. En suivant l'approche de Hill (1987); Demko et Hill (1988); Markakis et Psomas (2011), la garantie de l'agent  $i$  dépend

de  $\alpha_i$ , défini par l'utilité maximum de l'agent  $i$  pour un unique élément du matroïde.

Soit  $OPT_i(\mathcal{M})$  la valeur optimale d'une base qui maximise  $u_i$  pour  $i \in N$ . Nous supposons, sans perte de généralité, qu'après une mise à l'échelle, l'instance est normalisée de sorte que  $OPT_i(\mathcal{M}) = 1$  pour tout  $i \in N$ . Ainsi,  $\alpha_i$  est défini par  $\alpha_i = \max_{e \in X} \frac{u_i(e)}{OPT_i(\mathcal{M})} = \max_{e \in X} u_i(e)$  pour chaque agent  $i \in N$ .

Notre contribution est un protocole déterministe pour au plus 9 agents. Nous montrons que les agents peuvent bénéficier de la garantie de Markakis et Psomas (2011), qui est  $V_n(\alpha_i), \forall i \in N$ . Nous élaborons également des stratégies que les agents peuvent adopter s'ils veulent obtenir la garantie précédemment citée. Ces stratégies sont basées sur des algorithmes polynomiaux.

## 5.2 Un protocole pour deux agents

CUT-AND-CHOOSE est un protocole de partage sans envie pour deux agents lorsque la ressource est *divisible*. L'un des deux agents divise la ressource en deux parts qu'il juge égales, et l'autre agent choisit la moitié qu'il préfère. Nous proposons dans le protocole 5.1 un protocole similaire pour les matroïdes.

---

### Protocole 5.1 : DIVISER-ET-CHOISIR

---

**Données :** un matroïde  $\mathcal{M} = (X, \mathcal{F})$ , les agents 1 et 2.

- 1 L'agent 1 calcule une base  $A$  et une partition  $\{A_1, A_2\}$  de  $A$ .
  - 2 L'agent 2 choisit une part  $A_i \in \{A_1, A_2\}$  et la donne à l'agent 1. Puis, l'agent 2 complète  $A_i$  en ajoutant sa propre part  $B_{3-i}$  de sorte que  $A_i \cup B_{3-i}$  est une base de  $\mathcal{M}$ .
- 

L'agent 1 est appelé le *diviseur*. Montrons comment il peut se garantir une utilité d'au moins  $V_2(\alpha_1)$  en utilisant des algorithmes polynomiaux. L'agent 1 calcule une base  $A$  qu'il partitionne en deux parts  $A_1$  et  $A_2$  et sa contribution est l'une de ces parts. Ainsi, son utilité est d'au moins  $\min\{u_1(A_1), u_1(A_2)\}$ . Soit  $A^*$  une base qui maximise  $u_1$ , obtenue en appliquant le GLOUTON (algorithme 1.1, page 32). Nous avons  $u_1(A^*) = 1$  par hypothèse de normalisation. En utilisant le théorème 1.6 (page 37) avec  $n = 2$  et  $w = u_1$ , il existe une partition  $\{A_1^*, A_2^*\}$  de  $A^*$  telle que  $\min\{u_1(A_1^*), u_1(A_2^*)\} \geq \min\{u_1(A_1), u_1(A_2)\}$  pour toute base  $A$ . Ainsi, le diviseur ne perd jamais en partitionnant  $A^*$  en  $\{A_1^*, A_2^*\}$ . Une fois que  $A^*$  est calculée, on peut utiliser l'algorithme ALLOCATE de Markakis et Psomas (2011) (algorithme 2.1, page 65) qui prend en entrée : les éléments de  $A^*$  et deux agents fictifs, les deux ayant  $u_1$  comme fonction d'utilité pour les éléments. Le résultat est une bi-partition de  $A^*$  en  $\{A_1^*, A_2^*\}$  telle que  $\min\{u_1(A_1^*), u_1(A_2^*)\} \geq V_2(\alpha_1)$  car la plus grande utilité de l'agent 1 pour un élément de  $A^*$  est  $\alpha_1$ . Puisque les

algorithmes ALLOCATE et GLOUTON sont polynomiaux, alors le diviseur peut se garantir une utilité de  $V_2(\alpha_1)$  en temps polynomial.

L'agent 2 est appelé le *pivot* et il a en main la partition  $\{A_1, A_2\}$  communiquée par le diviseur. Montrons comment il peut se garantir une utilité de  $\frac{1}{2} \geq V_2(\alpha_2)$  en utilisant des algorithmes polynomiaux. Soit  $B^*$  une base qui maximise  $u_2$ . En utilisant le théorème 1.4 (page 27), il existe une partition de  $B^*$  en  $B_1^*$  et  $B_2^*$  telle que  $A_1 \cup B_2^*$  et  $A_2 \cup B_1^*$  sont deux bases de  $\mathcal{M}$ . Alors  $u_2(B^*) = u_2(B_1^*) + u_2(B_2^*) = 1$  en utilisant l'hypothèse de normalisation. Il est possible d'utiliser l'algorithme GLOUTON pour compléter  $A_1$  et  $A_2$  afin d'obtenir respectivement les bases  $A_1 \cup B_2$  et  $A_2 \cup B_1$  de  $\mathcal{M}$ .  $B_2$  et  $B_2^*$  sont des bases de  $\mathcal{M}/A_1$  mais  $B_2$  est optimale pour le matroïde  $\mathcal{M}/A_1$ . On obtient  $u_2(B_2) \geq u_2(B_2^*)$  et aussi  $u_2(B_1) \geq u_2(B_1^*)$ . D'où,  $\max\{u_2(B_1), u_2(B_2)\} \geq \max\{u_2(B_1^*), u_2(B_2^*)\} \geq \frac{1}{2}(u_2(B_1^*) + u_2(B_2^*)) = \frac{1}{2}$ .

**Proposition 5.1** *En utilisant le Protocole 5.1, Les agents 1 et 2 peuvent se garantir respectivement  $V_2(\alpha_1)$  et  $\frac{1}{2} = \frac{1}{2}V_1(2\alpha_2) \geq V_2(\alpha_2)$ .*

**Exemple 5.1** Soit le matroïde de partition  $\mathcal{M} = (X, \mathcal{F})$  (voir page 30) défini par  $X = \bigcup_{i=1}^3 X_i$  avec  $X_1 = \{e_1\}$ ,  $X_2 = \{e_2\}$ ,  $X_3 = \{e_3, e_4, e_5\}$  et  $\mathcal{F} = \{F \subseteq X : |F \cap X_1| \leq 1, |F \cap X_2| \leq 1, |F \cap X_3| \leq 2\}$ . Il y a 2 agents, leurs utilités sont données dans le tableau 5.1.

	$X_1$	$X_2$	$X_3$		
$e$	$e_1$	$e_2$	$e_3$	$e_4$	$e_5$
$u_1(e)$	0.4	0.25	0.2	0.15	0.15
$u_2(e)$	0.3	0.2	0.2	0.2	0.3

Tableau 5.1 – Exemple 5.1.

Appliquons DIVISER-ET-CHOISIR sur le matroïde  $\mathcal{M}$  et les agents 1 et 2. L'agent 1 applique GLOUTON  $(\mathcal{M}, u_1, \max)$  afin d'obtenir la base  $A = \{e_1, e_2, e_3, e_4\}$  qui maximise son utilité  $u_1(A) = OPT_1 = 1$ . Puis, il utilise ALLOCATE  $(A, \{1, 1\}, (u_1, u_1))$  (voir l'algorithme 2.1, page 65) pour partitionner  $A$  en deux parts  $\{A_1, A_2\}$ . L'agent 1 obtient  $A_1 = \{e_1\}$  et  $A_2 = \{e_2, e_3, e_4\}$  avec  $u_1(A_i) \geq V_2(\alpha_1) = \frac{1}{3}$ ,  $i = 1, 2$  car  $\alpha_1 = 0.4 \in NI(2, 1) = \left] \frac{1}{3}, \frac{2}{3} \right[$ .

Puis, l'agent 2 complète chaque part  $A_i$  par  $B_{3-i} := \text{GLOUTON}(\mathcal{M}/A_i, u_2, \max)$  pour  $i = 1, 2$ . Il obtient  $B_1 = \{e_1\}$  et  $B_2 = \{e_5, e_2, e_3\}$ . Il doit choisir une base parmi  $A_1 \cup B_2$  et  $A_2 \cup B_1$ . Puisque  $u_2(B_2) = 0.7 > u_2(B_1) = 0.3$  alors l'agent 2 choisit forcément de retourner la base  $A_1 \cup B_2$  où  $A_1$  est la part de l'agent 1 et  $B_2$  est la part de l'agent 2. On voit bien que  $u_1(A_1) = 0.4 > V_2(\alpha_1) = \frac{1}{3}$  et  $u_2(B_2) = 0.7 > \frac{1}{2} > V_2(\alpha_2) = 1 - 2\alpha_2 = 0.4$  car  $\alpha_2 = 0.3 \in I(2, 2) = \left[ \frac{3}{10}, \frac{1}{3} \right]$ .



### 5.3 Un protocole pour trois agents

Lorsqu'il y a un troisième agent, nous proposons un autre protocole décrit dans le protocole 5.2. Le pivot doit demander l'avis du troisième agent avant de réaliser toute action.

---

**Protocole 5.2 : DIVISER-INTERROGER-ET-CHOISIR**

---

**Données :** un matroïde  $\mathcal{M} = (X, \mathcal{F})$ , les agents 1, 2 et 3.

- 1 L'agent 1 calcule une base  $A$  qu'il partitionne en  $\{A_1, A_2, A_3\}$ .
  - 2 L'agent 2 choisit  $A_i$  pour un certain  $i \in \{1, 2, 3\}$  et propose à l'agent 3 de donner cette part à l'agent 1.
  - 3 L'agent 3 accepte :  $A_i$  est la contribution de l'agent 1. Appliquer DIVISER-ET-CHOISIR sur  $\mathcal{M}/A_i$  où l'agent 2 est le diviseur et l'agent 3 est le pivot.
  - 4 L'agent 3 refuse : appliquer DIVISER-ET-CHOISIR sur  $\mathcal{M}/A_i$  où l'agent 1 est le diviseur et l'agent 2 le pivot. Soit  $F$  la contribution des agents 1 and 2. L'agent 3 complète  $F$  en ajoutant sa propre part.
- 

Encore une fois, l'agent 1 est appelé le *diviseur* et il peut se garantir une utilité de  $V_3(\alpha_1)$  avec des algorithmes polynomiaux. À l'étape 1 du protocole, il suffit de construire une base  $A$  qui maximise  $u_1$  avec le GLOUTON et la diviser en 3 parts avec ALLOCATE (algorithme 2.1, page 65). Trois sous-ensembles disjoints  $\{A_1, A_2, A_3\}$  sont obtenus tels que  $\min\{u_1(A_1), u_1(A_2), u_1(A_3)\} \geq V_3(\alpha_1)$  et  $A_1 \cup A_2 \cup A_3$  est une base de  $\mathcal{M}$ . À l'étape 4 de DIVISER-INTERROGER-ET-CHOISIR, l'agent 1 est le diviseur du matroïde contracté  $\mathcal{M}/A_i$  et il peut soumettre  $\{A_1, A_2, A_3\} \setminus \{A_i\}$ . Par construction, l'utilité de l'agent 1 est au moins  $V_3(\alpha_1)$ .

Concentrons-nous maintenant sur l'agent 2. À l'étape 2 du protocole, on suppose que l'agent 2 choisit  $i \in \{1, 2, 3\}$  de sorte que  $OPT_2(\mathcal{M}/A_i) \geq \frac{2}{3}OPT_2(\mathcal{M}) = \frac{2}{3}$ . En utilisant le théorème 1.7 (page 38),  $i$  existe et peut être obtenu en temps polynomial. À l'étape 3 du protocole, l'agent 2 applique DIVISER-ET-CHOISIR (protocole 5.1, page 126) sur  $\mathcal{M}/A_i$  en qualité de diviseur. Le paramètre révisé dans ce matroïde contracté est noté  $\tilde{\alpha}_2$  et  $\tilde{\alpha}_2 \leq \frac{\alpha_2}{OPT_2(\mathcal{M}/A_i)} \leq \frac{3\alpha_2}{2}$  par hypothèse. L'agent 2 peut calculer une base  $B$  de  $\mathcal{M}/A_i$  vérifiant  $u_2(B) = OPT_2(\mathcal{M}/A_i)$  avec le GLOUTON. Il peut utiliser ALLOCATE afin de partitionner  $B$  en  $\{B_1, B_2\}$  de sorte que  $\min\{u_2(B_1), u_2(B_2)\} \geq V_2(\tilde{\alpha}_2)OPT_2(\mathcal{M}/A_i) \geq \frac{2}{3}V_2(\tilde{\alpha}_2)$ . Soit  $B_1$  ou  $B_2$  appartient à la base finale, donc l'utilité de l'agent 2 est d'au moins  $\frac{2}{3}V_2(\tilde{\alpha}_2)$ . En utilisant les faits que  $V_2$  est décroissante,  $\tilde{\alpha}_2 \leq \frac{3\alpha_2}{2}$  et le point I du Lemme 2.2 (page 66) avec  $n = 3$  et  $k = 2$ , on obtient  $\frac{2}{3}V_2(\tilde{\alpha}_2) \geq \frac{2}{3}V_2\left(\frac{3}{2}\alpha_2\right) \geq V_3(\alpha_2)$ ; donc l'utilité de l'agent 2 est au moins  $\frac{2}{3}V_2\left(\frac{3}{2}\alpha_2\right) \geq V_3(\alpha_2)$ . À l'étape 4 du protocole, l'agent 2 applique DIVISER-ET-CHOISIR (protocole 5.1, page 126) sur  $\mathcal{M}/A_i$  comme pivot. Nous savons de l'analyse de DIVISER-ET-CHOISIR que l'utilité du pivot est au moins l'utilité d'une base maximum, divisée par 2. Ainsi, l'utilité de l'agent 2 est

d'au moins  $\frac{1}{2}OPT_2(\mathcal{M}/A_i) \geq \frac{1}{2} \cdot \frac{2}{3} = \frac{1}{3}$  par hypothèse. En conclusion, l'agent 2 peut se garantir une utilité de  $\min \left\{ \frac{2}{3}V_2\left(\frac{3}{2}\alpha_2\right), \frac{1}{3} \right\} = \frac{2}{3}V_2\left(\frac{3}{2}\alpha_2\right)$  en utilisant le point II du Lemme 2.2 (page 66).

Pour l'agent 3, supposons qu'il accepte de donner  $A_i$  à l'agent 1 si et seulement si  $OPT_3(\mathcal{M}/A_i) \geq \frac{2}{3}$  (nous expliquerons plus loin pourquoi dans l'hypothèse 5.1, page 135). À l'étape 3 du protocole, l'agent 3 applique DIVISER-ET-CHOISIR sur  $\mathcal{M}/A_i$  comme pivot. L'utilité du pivot est au moins l'utilité d'une base optimale divisée par 2. Dans ce cas, l'utilité de l'agent 3 est au moins  $\frac{1}{2}OPT_3(\mathcal{M}/A_i) \geq \frac{1}{2} \cdot \frac{2}{3} = \frac{1}{3} = \frac{1}{3}V_1(3\alpha_3)$  par hypothèse. Notons que  $\frac{1}{3}V_1(3\alpha_3) \geq V_3(\alpha_3)$  grâce au point I du Lemme 2.2 (page 66). À l'étape 4 du protocole, l'agent 3 refuse de donner  $A_i$  à l'agent 1, donc  $OPT_3(\mathcal{M}/A_i) < \frac{2}{3}$  par hypothèse. L'agent 3 doit compléter  $F$ , une base de  $\mathcal{M}/A_i$  communiquée par les autres agents, avec  $F_3$  de sorte que  $F \cup F_3$  soit une base de  $\mathcal{M}$ .  $F \cup A_i$  est une base de  $\mathcal{M}$  et en utilisant le lemme 3.1 (page 90), nous savons que  $OPT_3(\mathcal{M}/F) \geq OPT_3(\mathcal{M}) - OPT_3(\mathcal{M}/A_i) \geq 1 - \frac{2}{3} = \frac{1}{3}$ . Ainsi, l'utilisation du GLOUTON mène à l'ensemble  $F_3$  évalué au moins  $\frac{1}{3}$  par l'agent 3.

**Proposition 5.2** *En utilisant le protocole 5.2, les agents 1, 2 et 3 peuvent se garantir respectivement  $V_3(\alpha_1)$ ,  $\frac{2}{3}V_2\left(\frac{3}{2}\alpha_2\right) \geq V_3(\alpha_2)$  et  $\frac{1}{3} = \frac{1}{3}V_1(3\alpha_3) \geq V_3(\alpha_3)$ .*

## 5.4 Un protocole général pour $n \leq 9$ agents

Nous présentons un protocole général pour  $n \leq 9$  agents dans le protocole 5.3, qui est une généralisation des protocoles 5.1 et 5.2.

---

**Protocole 5.3 : PROTOCOLE**

---

**Données** : un matroïde  $\mathcal{M} = (X, \mathcal{F})$ , un ensemble  $N = \{1, \dots, n\}$  de  $n \in \{1, 2, \dots, 9\}$  agents, un diviseur  $i_1 \in N$  qui est (si possible) le même que celui du précédent appel,  $n$  parts  $\{A_1, \dots, A_n\}$  fournies par le diviseur  $i_1$ .

**Résultat** : une base de  $\mathcal{M}$ .

- 1 **si**  $n = 1$  **alors** affecter la part  $A_1$  à l'agent  $i_1$  et **retourner**  $A_1$ .
  - 2 choisir un agent  $i_2 \in N \setminus \{i_1\}$  (le pivot), alors  $i_2$  renomme les  $n$  parts  $\{A_1, \dots, A_n\}$  (il produit une permutation  $\sigma$  de  $N$ ) telle que  $\forall k \in \{1, \dots, n-2\}$ , l'agent  $i_2$  accepte de partager le matroïde  $\mathcal{M}/(\cup_{j \leq k} A_{\sigma(j)})$  avec  $n-k-1$  autres agents.
  - 3 **si**  $n = 2$  **alors** affecter la part  $A_{\sigma(1)}$  à  $i_1$  et **retourner**  $A_{\sigma(1)} \cup \text{DIVISER}(\mathcal{M}/A_{\sigma(1)}, i_2, 1)$ .
  - 4  $N_0 \leftarrow N$  et pour  $k=1, \dots, n-2$ , soit  $N_k \subseteq N \setminus \{i_1, i_2\}$  l'ensemble des agents qui acceptent de partager le matroïde  $\mathcal{M}/(\cup_{j \leq k} A_{\sigma(j)})$  avec  $n-k-1$  autres agents.
  - 5  $k_0 \leftarrow \min\{k \geq 1 : |N_k| \geq n-k-2\}$ .
  - 6 **si**  $|N_{k_0}| \leq n-k_0-1$  **alors**
    - 7 **si**  $|N_{k_0}| = n-k_0-2$  **alors**  $J_0 \leftarrow N_{k_0} \cup \{i_1, i_2\}$  **sinon**  $J_0 \leftarrow N_{k_0} \cup \{i_2\}$ .
    - 8  $B_{J_0} \leftarrow \text{PROTOCOLE}(\mathcal{M}/(\cup_{j \leq k_0} A_{\sigma(j)}), J_0, i_1$  **si**  $i_1 \in J_0$  **et**  $i_2$  **sinon**,  $\{A_{\sigma(j)}, k_0+1 \leq j \leq n\}$  **si**  $i_1 \in J_0$  **et**  $\text{DIVISER}(\mathcal{M}/(\cup_{j \leq k_0} A_{\sigma(j)}), i_2, n-k_0)$  **sinon**).
    - 9 **retourner**  $B_{J_0} \cup \text{PROTOCOLE}(\mathcal{M}/B_{J_0}, N \setminus J_0, i_1$  **si**  $i_1 \in N \setminus J_0$  **et**  $i'_1 \in N \setminus J_0$  **sinon**,  $\{A_{\sigma(j)}, j \leq k_0\}$  **si**  $i_1 \in N \setminus J_0$  **et**  $\text{DIVISER}(\mathcal{M}/B_{J_0}, i'_1, k_0)$  **sinon**).
  - 10 choisir un ensemble  $J_1$  de  $n-k_0$  agents comme suit : au début  $J_1 = \{i_2\}$ , puis ajouter de nouveaux agents de  $N_{k_0} \cap N_{k_0-1}$ , puis ajouter de nouveaux agents de  $N_{k_0} \cap N_{k_0-2}$ , et ainsi de suite jusqu'à  $N_{k_0} \cap N_0$ .
  - 11  $B_{J_1} \leftarrow \text{PROTOCOLE}(\mathcal{M}/(\cup_{j \leq k_0} A_{\sigma(j)}), J_1, i_2, \text{DIVISER}(\mathcal{M}/(\cup_{j \leq k_0} A_{\sigma(j)}), i_2, n-k_0))$ .
  - 12 **si**  $\exists p \in \{1, 2\}$  tel que  $|N_{k_0-p} \setminus N_{k_0}| = p$  ou  $|N_{k_0-p} \setminus N_{k_0}| = p-1$  **alors**
    - 13  $p_0 \leftarrow \min\{p \in \{1, 2\} : |N_{k_0-p} \setminus N_{k_0}| = p \text{ ou } |N_{k_0-p} \setminus N_{k_0}| = p-1\}$ .
    - 14 **si**  $|N_{k_0-p_0} \setminus N_{k_0}| = p_0$  **alors**  $J_2 \leftarrow N_{k_0-p_0} \setminus N_{k_0}$  **sinon**  $J_2 \leftarrow N_{k_0-p_0} \setminus N_{k_0} \cup \{i_1\}$ .
    - 15  $B_{J_2} \leftarrow \text{PROTOCOLE}(\mathcal{M}/(\cup_{j \leq k_0-p_0} A_{\sigma(j)} \cup B_{J_1}), J_2, i_1$  **si**  $i_1 \in J_2$  **et**  $i'_1 \in J_2$  **sinon**,  $\{A_{\sigma(j)}, k_0-p_0+1 \leq j \leq k_0\}$  **si**  $i_1 \in J_2$  **et**  $\text{DIVISER}(\mathcal{M}/(\cup_{j \leq k_0-p_0} A_{\sigma(j)} \cup B_{J_1}), i'_1, p_0)$  **sinon**).
    - 16 **retourner**  $B_{J_1} \cup B_{J_2} \cup \text{PROTOCOLE}(\mathcal{M}/(B_{J_1} \cup B_{J_2}), N \setminus (J_1 \cup J_2), i_1$  **si**  $i_1 \in N \setminus (J_1 \cup J_2)$  **et**  $i'_1 \in N \setminus (J_1 \cup J_2)$  **sinon**,  $\{A_{\sigma(j)}, 1 \leq j \leq k_0-p_0\}$  **si**  $i_1 \in N \setminus (J_1 \cup J_2)$  **et**  $\text{DIVISER}(\mathcal{M}/(B_{J_1} \cup B_{J_2}), i'_1, k_0-p_0)$  **sinon**).
  - 17 **sinon si**  $|N_{k_0-2} \setminus N_{k_0}| = 0$  **alors**
    - 18  $J_3 = \{i_3\} \subset N_{k_0-1} \setminus N_{k_0}$  **et**  $B_{J_3} \leftarrow \text{DIVISER}(\mathcal{M}/(\cup_{j \leq k_0-1} A_{\sigma(j)} \cup B_{J_1}), i_3, 1)$ .
    - 19 **si**  $N_{k_0-2} \subset J_1$  **alors**  $J_4 = \{i_1\}$  **sinon**  $J_4 = \{i_4\} = N_{k_0-2} \setminus J_1$ .
    - 20 **si**  $J_4 = \{i_4\}$  **alors**  $B_{J_4} \leftarrow \text{DIVISER}(\mathcal{M}/(\cup_{j \leq k_0-2} A_{\sigma(j)} \cup B_{J_1} \cup B_{J_3}), i_4, 1)$  **sinon**  $B_{J_4} \leftarrow A_{\sigma(k_0-1)}$ .
    - 21 **retourner**  $B_{J_1} \cup B_{J_3} \cup B_{J_4} \cup \text{PROTOCOLE}(\mathcal{M}/(B_{J_1} \cup B_{J_3} \cup B_{J_4}), N \setminus (J_1 \cup J_3 \cup J_4), i_1$  **si**  $i_1 \in N \setminus (J_1 \cup J_3 \cup J_4)$  **et**  $i'_1 \in N \setminus (J_1 \cup J_3 \cup J_4)$  **sinon**,  $\{A_{\sigma(j)}, 1 \leq j \leq k_0-2\}$  **si**  $i_1 \in N \setminus (J_1 \cup J_3 \cup J_4)$  **et**  $\text{DIVISER}(\mathcal{M}/(B_{J_1} \cup B_{J_3} \cup B_{J_4}), i'_1, k_0-2)$  **sinon**).
  - 22 **sinon si**  $|N_{k_0-3} \setminus N_{k_0}| \leq 2$  **alors**
    - 23  $J_5 = \{i_5, i'_5\} \subset N_{k_0-2} \setminus N_{k_0}$  avec un nombre maximum d'éléments de  $N_{k_0-3} \setminus N_{k_0}$ .
    - 24  $B_{J_5} \leftarrow \text{PROTOCOLE}(\mathcal{M}/(\cup_{j \leq k_0-2} A_{\sigma(j)} \cup B_{J_1}), J_5, i'_1 \in J_5, \text{DIVISER}(\mathcal{M}/(\cup_{j \leq k_0-2} A_{\sigma(j)} \cup B_{J_1}), i'_1, 2))$ .
    - 25 **si**  $N_{k_0-3} \subset (J_1 \cup J_5)$  **alors**  $J_6 = \{i_1\}$  **sinon**  $J_6 = \{i_6\} = N_{k_0-3} \setminus (J_1 \cup J_5)$ .
    - 26 **si**  $J_6 = \{i_6\}$  **alors**  $B_{J_6} \leftarrow \text{DIVISER}(\mathcal{M}/(\cup_{j \leq k_0-3} A_{\sigma(j)} \cup B_{J_1} \cup B_{J_5}), i_6, 1)$  **sinon**  $B_{J_6} \leftarrow A_{\sigma(k_0-2)}$ .
    - 27 **retourner**  $B_{J_1} \cup B_{J_5} \cup B_{J_6} \cup \text{PROTOCOLE}(\mathcal{M}/(B_{J_1} \cup B_{J_5} \cup B_{J_6}), N \setminus (J_1 \cup J_5 \cup J_6), i_1$  **si**  $i_1 \in N \setminus (J_1 \cup J_5 \cup J_6)$  **et**  $i'_1 \in N \setminus (J_1 \cup J_5 \cup J_6)$  **sinon**,  $\{A_{\sigma(j)}, 1 \leq j \leq k_0-3\}$  **si**  $i_1 \in N \setminus (J_1 \cup J_5 \cup J_6)$  **et**  $\text{DIVISER}(\mathcal{M}/(B_{J_1} \cup B_{J_5} \cup B_{J_6}), i'_1, k_0-3)$  **sinon**).
  - 28 **sinon**
    - 29  $J_7 = \{i_7, i'_7, i''_7\} = N_{k_0-3} \setminus N_{k_0}$ .
    - 30  $B_{J_7} \leftarrow \text{PROTOCOLE}(\mathcal{M}/(\cup_{j \leq k_0-3} A_{\sigma(j)} \cup B_{J_1}), J_7, i'_1 \in J_7, \text{DIVISER}(\mathcal{M}/(\cup_{j \leq k_0-3} A_{\sigma(j)} \cup B_{J_1}), i'_1, 3))$ .
    - 31 **retourner**  $B_{J_1} \cup B_{J_7} \cup \text{PROTOCOLE}(\mathcal{M}/(B_{J_1} \cup B_{J_7}), N \setminus (J_1 \cup J_7), i_1, \{A_{\sigma(j)}, 1 \leq j \leq k_0-4\})$ .
-

L'idée générale du protocole 5.3 est la suivante : un premier agent, appelé le *diviseur* et noté  $i_1$ , trouve une base qu'il partitionne en  $n$  parts. Sa tâche est décrite dans l'algorithme 5.4 appelé **DIVISER** (page 134). Un deuxième agent, appelé le *pivot* et noté  $i_2$ , détermine un ordre (permutation)  $\sigma$  des  $n$  parts. Puis le protocole consiste à faire des appels récursifs sur une partition de l'ensemble des agents en au moins deux sous-instances appelées *début* et *fin*. Le protocole peut également affecter quelques agents à des sous-instances appelées *milieu*. Pour la sous-instance *début*, le protocole trouve un  $k \in \{1, \dots, n - 2\}$  tel que le matroïde contracté des  $k$  premières parts  $\sigma(1), \dots, \sigma(k)$  est partagé par  $n - k$  agents. Dans les sous-instances du *milieu*, le protocole peut affecter au plus 3 agents à une ou deux sous-instances du *milieu*. À la *fin*, le protocole affecte les agents restants à la sous-instance *fin*. Cela donne une solution partielle qui est complétée en une base. Des appels récursifs du protocole permettent d'effectuer un partage sur les compléments de *début*, éventuellement les instances du *milieu* et *fin*.

Notons que si l'on fait un appel récursif sur un sous-ensemble d'agents  $S \subseteq N$  de sorte que le précédent diviseur  $i_1$  est dans  $S$ , alors il est inutile de choisir un autre diviseur. Nous pouvons également garder les parts  $A_1, \dots, A_n$  que  $i_1$  a précédemment construit.

Le résultat principal de cette section est le théorème 5.1 que nous prouverons par la suite.

**Théorème 5.1.** *Si  $n \leq 9$ , alors le protocole 5.3 appliqué à  $\mathcal{M} = (X, \mathcal{F})$ ,  $N = \{1, \dots, n\}$ , un diviseur  $i_1 \in N$  et  $\{A_1, \dots, A_n\} = \text{DIVISER}(\mathcal{M}, i_1, n)$ , retourne une base  $B = \{B_1, \dots, B_n\}$  telle que  $u_i(B_i) \geq V_n(\alpha_i)$  pour tout  $i \in N$ .*

L'idée générale du Protocole 5.3 est décrite à travers une arborescence représentée dans la figure 5.1.

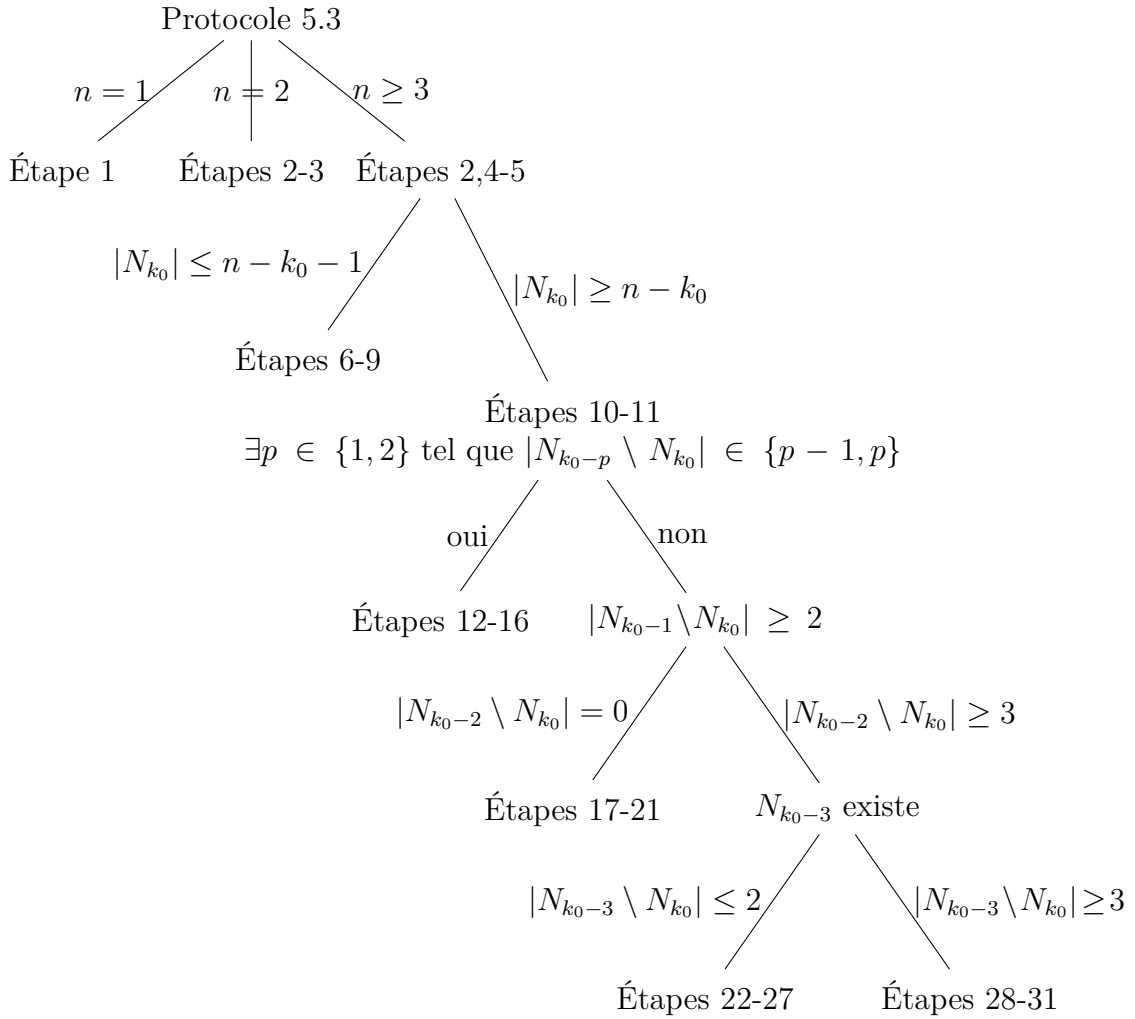


FIGURE 5.1 – Arborescence du Protocole 5.3.

Considérons l'exemple suivant sur l'application du Protocole 5.3.

**Exemple 5.2** Soient un matroïde  $\mathcal{M} = (X, \mathcal{F})$  et un ensemble d'agents  $N = \{1, \dots, 9\}$ . Soient  $i_1 = 1$  le diviseur et  $i_2 = 2$  le pivot (l'agent qui construit  $\sigma$ ) (voir le niveau 1 sur la figure 5.2, page 133).

Supposons qu'à l'étape 4 du Protocole 5.3, on obtient  $N_0 = N$ ,  $N_1 = \{3, 4\}$ ,  $N_2 = \{5, 6, 7, 9\}$ ,  $N_3 = \{7, 8, 9\}$ ,  $N_4 = \{7, 8\}$  et  $N_5 = \{3, 4, 5, 6\}$ . Nous nous arrêtons à  $N_5$  car  $|N_k| < n - k - 2$  pour  $k = 1, 2, 3, 4$  et  $|N_5| \geq n - 5 - 2 = 2$ , donc  $k_0 = 5$ .

Nous passons à l'étape 10 puisque  $|N_5| = 4 > n - 5 - 1 = 3$  et on conclut que  $J_1 = \{2, 5, 6, 3\}$ . Alors, à l'étape 11, les agents de  $J_1$  appliquent le protocole

5.3 pour partager le matroïde  $\mathcal{M}/(\cup_{j \leq 5} A_{\sigma(j)})$  où l'agent 2 est le diviseur et ils obtiennent  $B_{J_1}$  (voir le niveau 2 sur la figure 5.2). Nous n'imposons pas que l'agent 2 soit le diviseur, ce dernier doit juste être dans  $J_1$ .

Maintenant, nous passons à l'étape 22 car  $|N_{k_0-3} \setminus N_{k_0}| = |N_2 \setminus N_5| = 2$ . On obtient  $J_5 = \{7, 9\}$  et à l'étape 24, les agents de  $J_5$  appliquent le protocole 5.3 afin de partager le matroïde  $\mathcal{M}/(\cup_{j \leq 3} A_{\sigma(j)} \cup B_{J_1})$  où par exemple l'agent 7 est le diviseur et ils obtiennent  $B_{J_5}$  (voir le niveau 3 sur la figure 5.2). À l'étape 25,  $N_{k_0-3} = N_2 \subset (J_1 \cup J_5)$  alors  $J_6 = \{1\}$  et la part de l'agent 1 est  $B_{J_6} = A_{\sigma(k_0-2)} = A_{\sigma(3)}$  grâce à l'étape 26 (voir le niveau 4 sur la figure 5.2).

À l'étape 27, les agents restants  $N \setminus (J_1 \cup J_5 \cup J_6) = \{4, 8\}$  partagent le matroïde  $\mathcal{M}/(B_{J_1} \cup B_{J_5} \cup B_{J_6})$  et ils obtiennent  $B \setminus (B_{J_1} \cup B_{J_5} \cup B_{J_6})$  où  $B$  est la base résultante (voir le niveau 5).

Niveau 1	$A_{\sigma(1)}$	...	$A_{\sigma(k_0)}$	...	$A_{\sigma(n)}$	
Niveau 2	$A_{\sigma(1)}$	$A_{\sigma(2)}$	$A_{\sigma(3)}$	$A_{\sigma(4)}$	$A_{\sigma(5)}$	$B_{J_1}$ $J_1 = \{2, 5, 6, 3\}$
Niveau 3	$A_{\sigma(1)}$	$A_{\sigma(2)}$	$A_{\sigma(3)}$	$B_{J_5}$	$B_{J_1}$	$J_5 = \{7, 9\}$
Niveau 4	$A_{\sigma(1)}$	$A_{\sigma(2)}$	$B_{J_6}$	$B_{J_5}$	$B_{J_1}$	$J_6 = \{1\}$
Niveau 5	$B \setminus (B_{J_1} \cup B_{J_5} \cup B_{J_6})$		$B_{J_6}$	$B_{J_5}$	$B_{J_1}$	$N \setminus (J_1 \cup J_5 \cup J_6) = \{4, 8\}$

FIGURE 5.2 – Exemple d'exécution du Protocole 5.3.

Nous prouvons d'abord que le protocole 5.3 est bien défini, c'est-à-dire qu'une instruction **retourner** apparaît dans tous les cas.

**Lemme 5.1** *Si  $n = |N| \leq 9$  alors le protocole 5.3 est bien défini.*

Le preuve du lemme 5.1 est fournie dans l'annexe D.1, page 187.

### 5.4.1 Le point de vue du diviseur

Supposons que l'agent  $i_1 \in N$  est le diviseur et montrons comment il peut se garantir une utilité de  $V_n(\alpha_{i_1})$ .

**Proposition 5.3** *En utilisant le protocole 5.3, l'agent  $i_1$  (le diviseur) peut se garantir  $V_n(\alpha_{i_1})$  en temps polynomial.*

**Preuve.** Il est demandé au diviseur de produire une base  $A$  et de la partitionner en  $n$  parts  $A_1, \dots, A_n$ . Le protocole 5.3 est fait de façon à ce que la contribution de  $i_1$  soit l'une de ces parts. En effet, soit une part  $A_q$  (selon l'itération,  $q \in \{1, \sigma(1), \sigma(k_0 - 1), \sigma(k_0 - 2)\}$ ) est affectée à l'agent  $i_1$ , soit  $i_1$  reste le diviseur d'une plus petite instance composée d'un matroïde contracté  $\mathcal{M}'$  et de  $k$  agents (selon l'itération,  $k \in \{n - k_0, k_0, p_0, k_0 - p_0, k_0 - 2, k_0 - 3, k_0 - 4\}$ ). De plus,  $k$  parts  $A_{j_1}, \dots, A_{j_k} \in \{A_1, \dots, A_n\}$  sont incluses dans  $\mathcal{M}'$  et  $\cup_{\ell=1}^k A_{j_\ell}$  est une base de  $\mathcal{M}'$ . Par construction du protocole, l'agent  $i_1$  continue à proposer ces  $k$  parts. Par induction, l'agent  $i_1$  reçoit l'une de ces  $k$  parts  $\{A_{j_1}, \dots, A_{j_k}\}$ .

Par ailleurs, il est dans l'intérêt de l'agent  $i_1$  de maximiser  $\min_{i \in N} u_{i_1}(A_i)$ . Le théorème 1.6 (page 37) garantit que l'agent  $i_1$  peut choisir une base  $A^*$  maximum pour  $u_{i_1}$  car il existe une partition  $\{A_1^*, \dots, A_n^*\}$  de  $A^*$  vérifiant  $\min_{i \in N} u_{i_1}(A_i^*) \geq \min_{i \in N} u_{i_1}(A_i)$ . La base  $A^*$  peut être construite en temps polynomial avec le GLOUTON (algorithme 1.1, page 32) et nous savons que  $u_{i_1}(A^*) = 1$  par l'hypothèse de normalisation. Soit  $\beta$  l'utilité maximum de l'agent  $i_1$  pour un élément de  $A^*$ , c'est-à-dire  $\beta = \max_{e \in A^*} u_{i_1}(e)$ . Par construction de  $A^*$ ,  $\beta = \alpha_{i_1}$  puisqu'un élément de poids maximum est toujours choisi en premier par le GLOUTON. En utilisant l'algorithme ALLOCATE de Markakis et Psomas (2011) (algorithme 2.1, page 65), on peut trouver une partition de  $A^*$  vérifiant  $\min_{i \in N} u_{i_1}(A_i^*) \geq V_n(\alpha_{i_1})u_{i_1}(A^*) = V_n(\alpha_{i_1})$ .

Notons  $B_{i_1} \in \{A_1^*, \dots, A_n^*\}$  la contribution finale de l'agent  $i_1$ . Alors  $u_{i_1}(B_{i_1}) \geq \min_{i \in N} u_{i_1}(A_i^*) \geq V_n(\alpha_{i_1})$ . ■

En résumé, la façon dont le diviseur peut initialement se garantir  $V_n(\alpha_{i_1})$  est décrite dans l'algorithme 5.4.

---

**Algorithme 5.4 : DIVISER**

---

**Données :** un matroïde pondéré  $\mathcal{M} = (X, \mathcal{F}, u_i)$ , un agent  $i \in N$ , un entier  $n \geq 1$ .

- 1 L'agent  $i$  calcule une base  $A$  de  $\mathcal{M}$  maximum pour  $u_i$  en utilisant le GLOUTON.
  - 2 Normaliser les utilités des éléments de  $A$ , c'est-à-dire  $u_i(e) \leftarrow \frac{u_i(e)}{u_i(A)}$  pour tout  $e \in A$ .
  - 3 L'agent  $i$  partitionne  $A$  en  $n$  parts  $\{A_1, \dots, A_n\}$  en utilisant ALLOCATE où les éléments sont ceux de  $A$  et il y a  $n$  agents fictifs qui ont tous la même utilité  $u_i$ .
  - 4 retourner  $\{A_1, \dots, A_n\}$
- 

### 5.4.2 Les points de vue des non-diviseurs pour $n \geq 3$ agents

Dans le protocole 5.3, on demande à chaque non-diviseur  $i \in N \setminus \{i_1\}$  s'il accepte de partager  $\mathcal{M}$  contracté sur certaines parts fournies par le diviseur. Afin de

montrer comment un non-diviseur peut obtenir une garantie sur son utilité, nous faisons l'hypothèse suivante.

**Hypothèse 5.1** *Étant donné  $S \subset N$ , un agent  $i$  accepte de partager une base de  $\mathcal{M}/(\cup_{j \in S} A_{\sigma(j)})$  avec  $n - |S| - 1$  autres agents si et seulement si*

$$OPT_i(\mathcal{M}/(\cup_{j \in S} A_{\sigma(j)})) \geq \frac{n - |S|}{n} OPT_i(\mathcal{M})$$

**Observation 5.1** *Si l'agent  $i$  refuse de partager une base de  $\mathcal{M}/(\cup_{j \in S} A_{\sigma(j)})$  avec  $n - |S| - 1$  autres agents alors il accepte de partager une base de  $\mathcal{M}/(\cup_{j \in N \setminus S} A_{\sigma(j)})$  avec  $|S| - 1$  autres agents. En effet, on peut utiliser le lemme 3.1 (page 90) afin de montrer que  $OPT_i(\mathcal{M}/(\cup_{j \in N \setminus S} A_{\sigma(j)})) \geq \frac{|S|}{n} OPT_i(\mathcal{M})$ .*

Le protocole est fait de sorte que  $n - k$  agents partagent  $\mathcal{M}$  dans lequel  $k$  parts du diviseur sont contractées. Une justification de l'hypothèse 5.1 est que chaque non-diviseur fait une estimation de son utilité pour la base résultante. Cette estimation consiste à s'assurer une utilité de  $\frac{1}{n}$  dans le cas d'un partage équitable de la meilleure base (selon le point de vue du non-diviseur) du matroïde contracté. S'il y a  $k$  agents qui veulent partager le même matroïde que soi, alors l'utilité d'une base de ce matroïde doit être au moins  $\frac{k}{n}$ .

Après avoir trouvé l'ensemble d'agents approprié pour partager  $\mathcal{M}/(\cup_{j \in S} A_{\sigma(j)})$ , le prochain lemme montre que chaque agent de cet ensemble satisfait l'hypothèse 5.1 dans le protocole 5.3.

**Lemme 5.2** *Si  $n \leq 9$  alors à chaque appel récursif du Protocole 5.3, un ensemble  $I'$  d'agents et un ensemble indépendant  $B_{I'} \in \mathcal{F}$  sont obtenus tels que  $OPT_i(\mathcal{M}/B_{I'}) \geq \frac{|I'|}{n} OPT_i(\mathcal{M})$  pour tout  $i \in I' \setminus \{i_1\}$ .*

La preuve du lemme 5.2 est proposée dans l'annexe D.2, page 188.

### 5.4.3 Preuve du théorème principal

Nous supposons que  $B = \{B_1, \dots, B_n\}$  est la base retournée par le protocole 5.3 où  $B_i$  est la contribution de l'agent  $i \in N$ .

**Théorème 5.1** *Si  $n \leq 9$ , alors le protocole 5.3 appliqué à  $\mathcal{M} = (X, \mathcal{F})$ ,  $N = \{1, \dots, n\}$ , un diviseur  $i_1 \in N$  et  $\{A_1, \dots, A_n\} = \text{Diviser}(\mathcal{M}, i_1, n)$ , retourne une base  $B = \{B_1, \dots, B_n\}$  telle que  $u_i(B_i) \geq V_n(\alpha_i)$  pour tout  $i \in N$ .*

**Preuve.** La preuve est par récurrence sur  $n$ . Plus précisément, nous montrons par récurrence que pour n'importe quel matroïde  $\mathcal{M}$  et pour n'importe quel ensemble



$N$  d'au plus 9 agents, on obtient :

$$u_i(B_i) \geq V_n \left( \frac{\alpha_i}{OPT_i(\mathcal{M})} \right) OPT_i(\mathcal{M}) \quad (5.1)$$

Si  $n = 1$  alors le résultat est vrai car  $OPT_i(\mathcal{M}) = 1$  par hypothèse.

Supposons que le résultat est vrai pour au plus  $n - 1 \leq 8$  agents et montrons que l'inégalité (5.1) est valide pour  $n$  agents. Soit  $i \in N$  et supposons que  $i \neq i_1$  car le résultat est vrai pour l'agent  $i_1$  grâce à la proposition 5.3 (page 134).

Par construction du protocole, il existe une itération où  $i \in I'$  et le protocole est appelé pour  $\mathcal{M}' = \mathcal{M}/B_{I'} = (X', \mathcal{F}')$  avec  $I'$  comme ensemble d'agents. En utilisant le lemme 5.2 (page 135), nous savons que pour tout  $j \in I'$ ,

$$OPT_j(\mathcal{M}') \geq \frac{|I'|}{n} OPT_j(\mathcal{M}) = \frac{|I'|}{n} \quad (5.2)$$

Puisque  $|I'| < |N|$ , l'hypothèse de récurrence fournit en particulier pour  $j = i$  :

$$u_i(B_i) \geq V_{|I'|} \left( \frac{\alpha'_i}{OPT_i(\mathcal{M}')} \right) OPT_i(\mathcal{M}')$$

où  $\alpha'_i = \max_{e \in X'} u_i(e)$ . Puisque  $(V_k)_{k \in [n]}$  est une famille de fonctions décroissantes,  $\alpha'_i \leq \alpha_i$ ,  $\frac{\alpha'_i}{OPT_i(\mathcal{M}')} \leq \frac{n\alpha_i}{|I'|}$  et en utilisant (5.2), on conclut que

$$u_i(B_i) \geq \frac{|I'|}{n} V_{|I'|} \left( \frac{n\alpha_i}{|I'|} \right) OPT_i(\mathcal{M}) \quad (5.3)$$

En utilisant le point I. du lemme 2.2 page 66, nous savons que  $\frac{|I'|}{n} V_{|I'|} \left( \frac{n\alpha_i}{|I'|} \right) \geq V_n(\alpha_i)$ . Ainsi, l'inégalité (5.3) donne  $u_i(B_i) \geq V_n \left( \frac{\alpha_i}{OPT_i(\mathcal{M})} \right) OPT_i(\mathcal{M})$  puisque  $OPT_i(\mathcal{M}) = 1$ .

En conclusion, la récurrence est démontrée et le résultat s'ensuit. ■

Nous pouvons montrer un résultat plus fort sur l'utilité de la contribution de chaque agent lorsque  $n \leq 9$  comme pour les propositions 5.1 (page 127) et 5.2 (page 129) respectivement pour 2 et 3 agents.

**Théorème 5.2** *Si  $n \leq 9$ , alors il existe une permutation  $\pi$  de  $\{1, \dots, n\}$  telle que le protocole retourne une base  $B = \{B_1, \dots, B_n\}$  avec*

$$\forall i \in N, u_i(B_i) \geq \frac{\pi(i)}{n} V_{\pi(i)} \left( \frac{n\alpha_i}{\pi(i)} \right)$$

**Preuve.** Comme pour le théorème 5.1, nous montrons par récurrence sur  $n \leq 9$  que pour tout matroïde  $\mathcal{M}$  et pour tout ensemble  $N$  d'au plus 9 agents, il existe une permutation  $\pi$  de  $\{1, \dots, n\}$  où

$$\forall i \in N, u_i(B_i) \geq \frac{\pi(i)}{n} V_{\pi(i)} \left( \frac{n\alpha_i}{\pi(i)OPT_i(\mathcal{M})} \right) OPT_i(\mathcal{M}) \quad (5.4)$$

Si  $n = 1$ , alors  $\pi(1) = 1$  et le résultat est vrai car  $OPT_1(\mathcal{M}) = 1$ .

Supposons que le résultat est vrai pour au plus  $n - 1 \leq 8$  agents et montrons que l'inégalité (5.4) est valide pour  $n$  agents. Étudions la situation où le premier appel récursif est fait ; il existe une partition de  $N$  en :

- $J' = \{i_1\}$  et  $J'' = \{i_2\}$  à l'étape 3,
- $J' = J_0$  et  $J'' = N \setminus J_0$  à l'étape 9,
- $J' = J_1, J'' = J_2$  et  $J''' = N \setminus (J_1 \cup J_2)$  à l'étape 16,
- $J' = J_1, J'' = J_3, J''' = J_4$  et  $J'''' = N \setminus (J_1 \cup J_3 \cup J_4)$  à l'étape 21,
- $J' = J_1, J'' = J_5, J''' = J_6$  et  $J'''' = N \setminus (J_1 \cup J_5 \cup J_6)$  à l'étape 27,
- $J' = J_1, J'' = J_7$ , et  $J''' = N \setminus (J_1 \cup J_7)$  à l'étape 31.

Si durant le processus, une part  $A_p$  a été allouée à l'agent  $i_1$ , alors on pose  $\pi(i_1) = n$  et l'inégalité (5.4) est vérifiée pour l'agent  $i_1$  grâce au théorème 5.1 (page 135). Maintenant, supposons que l'agent  $i_1$  n'a pas encore reçu de part. Alors, en utilisant l'hypothèse de récurrence, il y a deux ou trois ou quatre permutations  $\pi', \pi'', \pi'''$  et  $\pi''''$  respectivement de  $\{1, \dots, |J'|\}$ ,  $\{1, \dots, |J''|\}$ ,  $\{1, \dots, |J''|\}$  et  $\{1, \dots, |J''''|\}$  et deux ou trois ou quatre matroïdes  $\mathcal{M}', \mathcal{M}'', \mathcal{M}'''$  et  $\mathcal{M}''''$  tels que l'inégalité (5.4) est valide. Par exemple, pour  $J'$  cela signifie qu'il existe une bijection de  $\{1, \dots, |J'|\}$  dans  $J'$  telle que

$$u_{f(i)}(B_{f(i)}) \geq \frac{\pi'(i)}{|J'|} V_{\pi'(i)} \left( \frac{|J'| \alpha_{f(i)}}{\pi'(i) OPT_{f(i)}(\mathcal{M}')} \right) OPT_{f(i)}(\mathcal{M}'), \quad \forall i \in \{1, \dots, |J'|\}$$

Nous savons grâce au lemme 5.2 (page 135) que  $OPT_{f(i)}(\mathcal{M}') \geq \frac{|J'|}{n} OPT_{f(i)}(\mathcal{M})$ . En l'associant à l'inégalité précédente, et en utilisant la décroissance de  $V_{\pi'(i)}$  et  $OPT_{f(i)}(\mathcal{M}) = 1$ , on obtient

$$u_{f(i)}(B_{f(i)}) \geq \frac{\pi'(i)}{|J'|} V_{\pi'(i)} \left( \frac{|J'| \alpha_{f(i)}}{\pi'(i) OPT_{f(i)}(\mathcal{M}')} \right) OPT_{f(i)}(\mathcal{M}') \geq \frac{\pi'(i)}{n} V_{\pi'(i)} \left( \frac{n\alpha_{f(i)}}{\pi'(i)} \right) \quad (5.5)$$

En utilisant le point II. du lemme 2.2 (page 66), nous avons pour tout  $n \geq 1$ ,

$$\frac{1}{n} V_1 \left( \frac{n}{1} \alpha_i \right) \geq \frac{2}{n} V_2 \left( \frac{n}{2} \alpha_i \right) \geq \dots \geq \frac{n}{n} V_n \left( \frac{n}{n} \alpha_i \right) \quad (5.6)$$

D'où, en utilisant les inégalités (5.5) et (5.6) et en posant par exemple  $\pi(i) = \pi'(i)$  pour  $i \in \{1, \dots, |J'|\}$ ,  $\pi(|J'| + i) = \pi''(i)$  pour  $i \in \{1, \dots, |J''|\}$ ,  $\pi(|J'| +$

$|J''| + i) = \pi'''(i)$  pour  $i \in \{1, \dots, |J''|\}$  et  $\pi(|J'| + |J''| + |J'''| + i) = \pi''''(i)$  pour  $i \in \{1, \dots, |J''''|\}$ , on obtient le résultat attendu. ■

Les inégalités du théorème 5.2 sont atteintes par le protocole lorsqu'à chaque étape,  $k_0 = 1$  et  $|N_{k_0}| = n - 2$ .

## Conclusion et perspectives

Le protocole présenté dans ce chapitre est valide pour  $n \leq 9$  agents. Nous avons tenté de produire un protocole pour  $n = 10$  agents mais le nombre de cas à vérifier augmente rapidement et nous n'avons pas pu écrire une version concise du protocole. Cependant, nous conjecturons l'existence d'un tel protocole pour n'importe quel nombre d'agents  $n$ . Il paraît nécessaire de changer la définition de  $J_1$  dans le protocole 5.3. Par exemple,  $J_1$  peut être construit en ajoutant d'abord  $i_2$ , puis les éléments de  $N_{k'} \cap N_{k_0}$  pour un certain  $k' \in \{1, \dots, k_0 - 1\}$  de sorte que  $N_{k'} \setminus J_1$  soit minimum, et ainsi de suite.

À défaut d'avoir un protocole général pour n'importe quel nombre d'agents  $n \geq 2$ , il serait intéressant de montrer l'existence d'une partition d'une base en  $n$  parts de sorte que l'utilité de chaque agent  $k$  pour sa part vaille au moins  $\frac{k}{n}V_k(\frac{n}{k}\alpha_k)$  pour tout  $k \in \{1, \dots, n\}$ .

Notons que nous utilisons dans ce chapitre la fonction  $V_n$  pour définir une nouvelle garantie. Alors que la fonction  $W_n$  (proposée dans le chapitre 4) est supérieure ou égale à  $V_n$  et est garantie pour les matroïdes. Le fait d'utiliser ici  $V_n$  au lieu de  $W_n$  est dû à la non-monotonie de  $W_n$  alors que nous avons besoin d'une fonction monotone dans la preuve du théorème 5.1. Cependant, l'agent le moins heureux qui est le diviseur  $i_1$  peut se garantir  $W_n(\alpha_{i_1})$  en appliquant l'algorithme SEUIL (algorithme 4.1, page 104) au lieu de l'algorithme ALLOCATE de Markakis et Psomas (2011) (algorithme 2.1, page 65) pour partitionner sa base en  $n$  parts.

Puisqu'il existe maintenant deux différentes garanties dans le pire cas du problème de la base d'un matroïde multi-agents :  $\frac{k}{n}V_k(\frac{n}{k}\alpha_k)$  pour chaque agent  $k \in \{1, \dots, n\}$  proposée dans ce chapitre, et  $W_n(\alpha_k)$  pour chaque agent  $k \in \{1, \dots, n\}$  proposée dans le chapitre 4, il serait judicieux de comparer ces deux garanties. Cependant, aucune n'est dominée par l'autre. En effet,  $\frac{n}{n}V_n(\frac{n}{n}x) = V_n(x) \leq W_n(x) \leq \frac{1}{n} = \frac{1}{n}V_1(\frac{n}{1}x)$  pour tout  $x \in [0, 1]$  et pour tout  $n \geq 2$ , mais pour  $k \in \{2, 3, \dots, n-1\}$  aucune garantie ne domine l'autre comme on peut le voir sur la figure 5.3 pour  $n = 3$ .

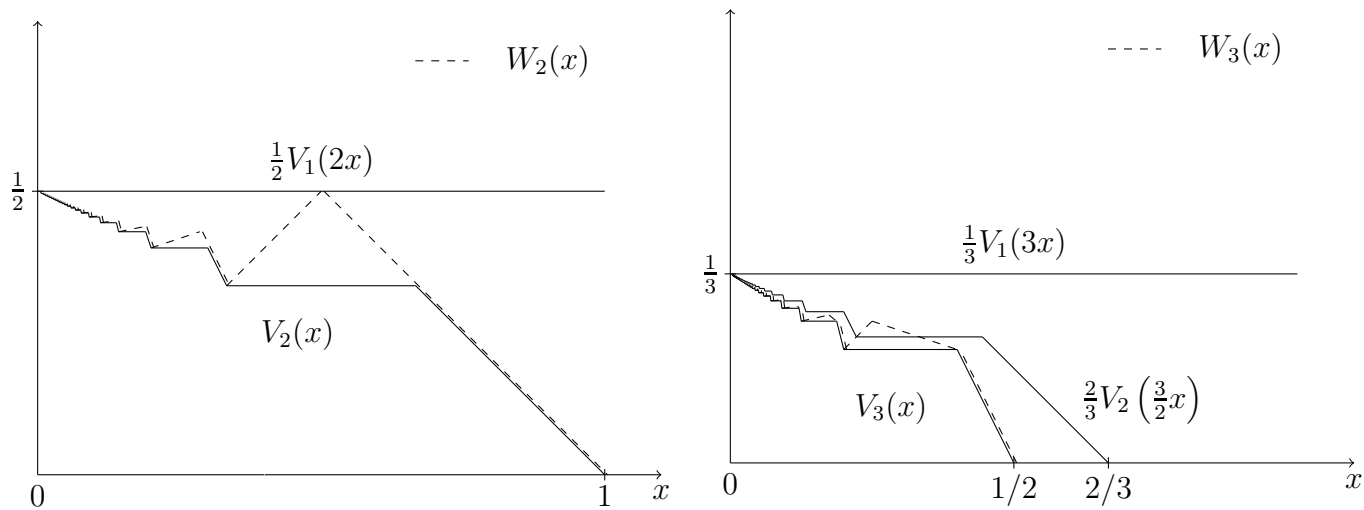


FIGURE 5.3 – Les fonctions  $\frac{k}{n}V_k\left(\frac{n}{k}x\right)$  et  $W_n(x)$  pour  $n \in \{2, 3\}$  et  $k \in \{1, \dots, n - 1\}$ .



# Solutions de compromis pour des matroïdes pondérés coloriés

Nous étudions une extension du problème traité dans les chapitres 3, 4 et 5 dans le cas de deux agents. L'objectif est de trouver une base commune à deux agents ayant des utilités de différents types. L'utilité du premier agent est additive, tandis que celle du second agent est une fonction sous-modulaire particulière, dite *coloriée pondérée*. Nous recherchons des solutions de compromis avec une garantie dans le pire cas sur l'utilité de chaque agent. Nous proposons des algorithmes déterministes polynomiaux pour plusieurs variantes du problème qui atteignent différentes garanties à rapports constants. Nous prouvons également que certaines garanties ne peuvent être atteintes.

Ce chapitre a fait l'objet d'un article à paraître dans la revue internationale *Discrete Applied Mathematics* (DAM). Une version antérieure apparaît dans les actes de la conférence internationale «The 20th European Conference on Artificial Intelligence (ECAI'12)» (Gourvès *et al.*, 2012).

## Sommaire

---

<b>Introduction . . . . .</b>	<b>124</b>
<b>5.1 Le modèle . . . . .</b>	<b>125</b>
<b>5.2 Un protocole pour deux agents . . . . .</b>	<b>126</b>
<b>5.3 Un protocole pour trois agents . . . . .</b>	<b>128</b>
<b>5.4 Un protocole général pour <math>n \leq 9</math> agents . . . . .</b>	<b>129</b>
5.4.1 Le point de vue du diviseur . . . . .	133
5.4.2 Les points de vue des non-diviseurs pour $n \geq 3$ agents .	134
5.4.3 Preuve du théorème principal . . . . .	135
<b>Conclusion et perspectives . . . . .</b>	<b>138</b>

---

## Introduction

Ce chapitre traite de l'existence et de l'obtention d'une solution  $s$  commune à deux agents. L'intérêt de l'agent  $i \in \{1, 2\}$  pour l'ensemble des solutions possibles est représenté par une fonction d'utilité  $u_i$ . Lorsque ces fonctions sont en conflit, il n'existe pas forcément de solution réalisable  $s$ , telle que  $u_1(s)$  et  $u_2(s)$  soient toutes les deux proches de l'optimum. Il paraît alors important de trouver un compromis. Si  $s_i^*$  représente la solution qui maximise  $u_i(s_i^*)$  alors il est possible qu'une solution de compromis  $s$  vérifie  $u_1(s)/u_1(s_1^*) \approx 1$  et  $u_2(s)/u_2(s_2^*) \approx 0$  (ou de façon équivalente  $u_1(s)/u_1(s_1^*) \approx 0$  et  $u_2(s)/u_2(s_2^*) \approx 1$ ). Ce cas pathologique indique que  $s$  peut ne pas être équitable, c'est-à-dire proche de l'optimum pour un agent, et très loin de l'optimum pour le second agent.

Pour quelle bornes inférieures de  $u_1(s)/u_1(s_1^*)$  et  $u_2(s)/u_2(s_2^*)$  une solution  $s$  est-elle garantie d'exister ? Quel algorithme peut faire face à des fonctions d'utilité qui peuvent être de différentes natures et telles qu'on peut en déduire *a priori* des bornes inférieures non triviales sur  $u_1(s)/u_1(s_1^*)$  et  $u_2(s)/u_2(s_2^*)$  ?

Dans ce chapitre, nous recherchons des algorithmes  $(\epsilon_1, \epsilon_2)$ -*approchés*, c'est-à-dire des algorithmes retournant une solution  $s$  telle que  $u_1(s)/u_1(s_1^*) \geq \epsilon_1$  et  $u_2(s)/u_2(s_2^*) \geq \epsilon_2$  pour toute instance. Cela représente une approximation du point idéal du problème bi-objectif qui consiste à maximiser les utilités des deux agents (voir la section 1.2.3, page 21). Les vecteurs  $(\epsilon_1, \epsilon_2)$  qui sont optimaux au sens de Pareto sont d'une importance particulière. Il existe plusieurs articles qui traitent des algorithmes  $(\epsilon_1, \epsilon_2)$ -*approchés*, incluant Hansen (1980); Warburton (1987); Stein et Wein (1997); Marathe *et al.* (1998); Papadimitriou et Yannakakis (2000); Vassilvitskii et Yannakakis (2005); Goel et Meyerson (2006); Escoffier *et al.* (2013).

Nous considérons des problèmes ayant une structure de *matroïde* où une solution réalisable est une base. Nous supposons que l'utilité du premier agent est additive et que l'utilité du second agent est plus complexe ; c'est une fonction sous-modulaire particulière qui modélise des préférences plus élaborées. Cette fonction sous-modulaire est plus connue dans la littérature sous le nom de fonction *pondérée coloriée* Hassin *et al.* (2007). Globalement, chaque élément  $e$  du matroïde a une couleur  $\mathcal{L}(e)$  parmi un ensemble de couleurs  $\{\ell_1, \dots, \ell_p\}$  et chaque couleur  $\ell$  a un gain positif  $g(\ell) \geq 0$ . Cette fonction pondérée coloriée mesure le gain global des couleurs utilisées dans la solution. Par exemple, si une solution  $\{e_1, e_2, e_3\}$  est telle que  $\mathcal{L}(e_1) = \mathcal{L}(e_2) = \ell_1$  et  $\mathcal{L}(e_3) = \ell_2$  avec  $g(\ell_1) = 2$  et  $g(\ell_2) = 1$ , alors son gain (pondéré colorié) est  $g(\ell_1) + g(\ell_2) = 3$  même si la couleur  $\ell_1$  est utilisée deux fois. Cette fonction est également connue sous le nom de *prix chromatique* dans Maffioli *et al.* (2007). Elle est utilisée par Hassin *et al.* (2007); Maffioli *et al.* (2007) pour plusieurs problèmes de connectivité dans les

graphes ou les matroïdes. Nous étudions également le cas particulier des couleurs *non pondérées*, c'est-à-dire  $g(\ell) = 1$  pour chaque couleur  $\ell$ . Dans ce cas, l'utilité du second agent tient compte du nombre de couleurs utilisées dans une solution. Plusieurs articles traitent de l'optimisation du nombre de couleurs pour les arbres couvrants, les chemins, les cycles hamiltoniens, les coupes, etc. (Broersma et Li, 1997; Krumke et Wirth, 1998; Brüggemann *et al.*, 2003; Broersma *et al.*, 2005; Couëtoux *et al.*, 2010; Zhang *et al.*, 2011; Tang et Zhang, 2012).

Un exemple typique des instances couvertes dans ce chapitre est le suivant :

**Exemple 6.1** Un touriste souhaite visiter une ville pendant  $n$  jours. Il fait appel à une agence de voyage pour organiser son séjour avec (au plus) une activité par jour. Il y a  $m$  activités disponibles :  $a_1, \dots, a_m$ . Chaque activité  $a_i$  a :

- une *couleur*  $\mathcal{L}(a_i)$  qui caractérise la catégorie de l'activité (par exemple visiter un monument, assister à un spectacle, manger dans un restaurant gastronomique, etc.); il peut y avoir plusieurs activités de même catégorie. Les couleurs constituent une partition de l'ensemble des activités ;
- un ensemble de jours  $D(a_i)$  durant lesquels  $a_i$  peut être programmée ;
- un poids positif  $w(a_i)$  indiquant le bénéfice de l'agence de voyage lorsque l'activité est programmée.

Chaque couleur  $\ell$  a un gain positif, appelé  $g(\ell)$ , qui représente l'intérêt du touriste pour toute activité  $a_i$  telle que  $\mathcal{L}(a_i) = \ell$ .

Si  $A$  représente le sous-ensemble d'activités programmées, alors l'agence de voyage souhaite maximiser son profit  $\sum_{a \in A} w(a)$ . Le point de vue du touriste est différent puisqu'il veut maximiser  $\sum_{\ell \in \{\mathcal{L}(a) : a \in A\}} g(\ell)$ . Nous supposons que l'utilité du touriste augmente lorsqu'il participe à une activité d'une nouvelle catégorie (couleur).

Le problème du touriste/agence-de-voyage peut être modélisé par le matroïde transversal (voir page 29). En effet, soit  $X = \{a_1, \dots, a_m\}$  l'ensemble des activités et  $X_i \subseteq X$  le sous-ensemble d'activités disponibles le jour  $i$ ,  $i = 1, \dots, n$ . Un ensemble d'activités  $T \subseteq X$  est réalisable s'il existe une application injective  $\Phi : T \rightarrow [1..n]$  qui affecte au plus une activité de  $X_i$  par jour  $i$  ( $i = 1, \dots, n$ ).

D'autres applications de certains cas particuliers de notre problème sont mentionnés par Maffioli *et al.* (2007).

En raison des conflits entre les agents sur le choix de la solution de compromis et de la difficulté technique liée au traitement de deux fonctions d'utilité de types différents, nous nous orientons vers des procédures de *choix séquentiel* (*picking sequences*) (Bouveret et Lang, 2011). Ces algorithmes opèrent de la façon suivante : en partant du vide, la solution est construite de manière séquentielle en ajoutant, à chaque étape, un élément à la solution jusqu'à obtenir une solution réalisable. Les éléments sont ajoutés par les agents en définissant, au préalable, l'ordre de



passage des agents. Par exemple, s'il y a 3 agents, alors la séquence 1132 signifie que l'agent 1 commencera par ajouter deux éléments successivement, puis l'agent 3 ajoutera un élément et enfin ce sera le tour de l'agent 2 d'ajouter un élément. Il existe plusieurs articles qui traitent des procédures de choix séquentiel pour le problème classique d'allocation de biens indivisibles dont Bouveret et Lang (2011); Kalinowski *et al.* (2013a,b); Bouveret et Lang (2014).

Le contexte du problème étudié dans ce chapitre est présenté dans la section 6.1 : la section 6.1.1 présente le modèle et la section 6.1.2 explique l'approche utilisée et les instances que nous manipulons. Dans la section 6.2, nous proposons un algorithme  $(1/2, 1/4)$ -approché (aussi une  $(1/3, 1/3)$ -approximation) appelé ALT-GLOUTON qui simule un processus naturel où deux agents construisent une solution commune. Ce résultat est une extension du résultat d'Escoffier *et al.* (2013) qui s'intéressent à construire un arbre couvrant commun (matroïde graphique) à deux agents ayant des fonctions d'utilité additives. Dans la section 6.3, nous considérons le sous-cas d'utilités additives correspondant au cas où chaque couleur apparaît exactement une fois. Nous présentons un algorithme  $(\frac{x}{x+y}, \frac{y}{x+y+xy})$ -approché pour tout  $x, y \in \mathbb{N}$  (non tous les deux nuls) lorsqu'une solution réalisable contient au moins  $p(x+y)$  éléments, pour un entier  $p > 0$ . Cet algorithme généralise ALT-GLOUTON pour des utilités additives. La complexité générale est mentionnée dans la section 6.4 ainsi qu'un algorithme qui obtient un optimum lexicographique particulier. Dans la section 6.5, nous étudions le cas des *couleurs non-pondérées* (correspondant à  $g(\ell_i) = 1$  pour tout  $i$ ). Nous montrons les résultats suivants : d'une part, nous produisons une  $(\frac{k-1}{k}, \frac{1}{k})$ -approximation en temps polynomial pour tout entier strictement positif  $k$  donné en entrée, et d'autre part, nous exhibons quelques instances sans aucune  $(\alpha, 1-\alpha)$ -approximation pour tout  $\alpha \in ]0, 1[$  avec  $\alpha \notin \{\frac{k-1}{k} : k \text{ est un entier positif}\}$ . Dans la section 6.5.2, nous étudions le problème d'obtention d'une base de poids maximum sous la contrainte qu'elle contienne exactement  $k$  couleurs. Nous fournissons quelques propriétés structurelles et un algorithme pour obtenir une telle base lorsqu'initialement il y a une unique base optimale pour l'agent 1. Quelques questions ouvertes sont indiquées dans la conclusion.

## 6.1 Contexte

### 6.1.1 Le modèle

Soient  $\mathcal{M} = (X, \mathcal{F})$  un matroïde et  $\{\ell_1, \dots, \ell_p\}$  un ensemble de  $p \geq 1$  couleurs. Considérons les fonctions suivantes :

- $w : X \rightarrow \mathbb{R}^+$ , où  $w(x)$  est le *poids* de  $x \in X$  ;

- $\mathcal{L} : X \rightarrow \{\ell_1, \dots, \ell_p\}$ , où  $\{\ell_1, \dots, \ell_p\}$  est l'ensemble des couleurs et  $\mathcal{L}(e)$  est la couleur de  $e$  ;
- $g : \{\ell_1, \dots, \ell_p\} \rightarrow \mathbb{R}^+$ , où  $g(\ell)$  est le *gain* de la couleur  $\ell$ .

Par abus de notation, nous écrivons souvent  $g(x)$  au lieu de  $g(\mathcal{L}(x))$  pour  $x \in X$ . Les couleurs de l'ensemble  $X' \subseteq X$  forment un ensemble noté  $\mathcal{L}(X')$  et défini par  $\bigcup_{x \in X'} \{\mathcal{L}(x)\}$ .

Les bases du matroïde  $\mathcal{M}$  sont les solutions réalisables du problème. Les utilités du premier et du second agent sont respectivement  $w$  et  $g$ . Un ensemble  $F \in \mathcal{F}$  est évalué par deux fonctions :  $u_1(F) = w(F) = \sum_{x \in F} w(x)$  et  $u_2(F) = g(F) = \sum_{l \in \mathcal{L}(F)} g(l)$  qui doivent être maximisées. Une instance est alors un quadruplet  $\langle \mathcal{M}, w, \mathcal{L}, g \rangle$ .

Comme nous l'avons mentionné plus tôt, le GLOUTON (algorithme 1.1, page 32) trouve une base  $B_1$  qui maximise  $w(B_1)$ . Trouver  $B_2$  qui maximise  $g(B_2)$  peut être fait en temps polynomial en recherchant une solution de coût maximum dans l'intersection de deux matroïdes (voir page 35). Le premier matroïde  $\mathcal{M} = (X, \mathcal{F})$  est le matroïde initial. Le second matroïde  $\mathcal{M}' = (X, \mathcal{F}')$  est un matroïde de partition (voir page 30) défini par les couleurs de  $X$  :  $\mathcal{F}' = \{F \subseteq X : |\mathcal{L}(F)| = |F|\}$ . Chaque élément  $x \in X$  a un coût défini par  $g(\mathcal{L}(x))$ . On utilise un algorithme approprié pour calculer un ensemble indépendant dans  $\mathcal{F} \cap \mathcal{F}'$  de coût maximum (voir Frank (1981); Korte et Vygen (2008); Schrijver (2003)) puis on le complète en une base de  $\mathcal{M}$  si nécessaire. Il n'est pas difficile de voir que la base résultante maximise  $g$ . La même approche est proposée par Maffioli *et al.* (2007) pour maximiser le nombre de couleurs d'une base. Leur résultat est une conséquence du théorème 6.4 (page 156) que nous montrerons dans la section 6.4 lorsque tous les éléments de  $X$  ont le même poids.

### 6.1.2 Approximation non triviale

Lorsqu'on se concentre sur l'agent  $i$ , on dit qu'un algorithme  $\mathcal{A}$  est  $\epsilon$ -approché si, pour toute instance,  $\mathcal{A}$  retourne une solution  $s$  vérifiant  $u_i(s)/u_i(s_i^*) \geq \epsilon$  pour tout  $i$ . Ici,  $s_i^*$  est une solution qui maximise  $u_i(s_i^*)$  et  $\epsilon \in [0, 1]$  est le *rapport d'approximation* ou la *garantie de performance* (Vazirani, 2001).

Évidemment, retourner une solution qui maximise  $w$  (resp.  $g$ ) donne une  $(1, 0)$ -approximation (resp.  $(0, 1)$ -approximation). Nous recherchons des réels  $\epsilon_1$  et  $\epsilon_2$  strictement positifs de façon à ce que la solution constitue un compromis non trivial. Les prochains exemples montrent, pour le modèle général considéré dans ce chapitre (décrit dans la section 6.1.1), qu'il n'existe pas de compromis non trivial  $(\epsilon_1, \epsilon_2)$ -approché (avec  $\epsilon_1 > 0$  et  $\epsilon_2 > 0$ ) si on considère toute la classe des matroïdes.

**Exemple 6.2** Considérons le matroïde  $(X = \{a, b\}, \mathcal{F} = \{\emptyset, \{a\}, \{b\}\})$  où  $w(a) = 1$ ,  $w(b) = 0$ ,  $\ell(a) = l$ ,  $\ell(b) = l'$ ,  $g(l) = 0$  et  $g(l') = 1$ . Le rang est 1.

**Exemple 6.3** Considérons le matroïde  $\mathcal{M} = (X, \mathcal{F})$  où  $X = \{a, b, c\}$  et  $\mathcal{F} = \{\emptyset, \{a\}, \{b\}, \{c\}, \{a, c\}, \{b, c\}\}$  avec  $w(a) = 1$ ,  $w(b) = w(c) = 0$ ,  $\ell(a) = \ell(c) = l$ ,  $\ell(b) = l'$ ,  $g(l) = 0$  et  $g(l') = 1$ .

Dans ces deux exemples, pour tout ensemble indépendant  $F$ ,  $\min\{w(F), g(F)\} = 0$  tandis que  $\max_{F \in \mathcal{F}}\{w(F)\} = \max_{F \in \mathcal{F}}\{g(F)\} = 1$ . Cela signifie que soit  $\epsilon_1 = 0$ , soit  $\epsilon_2 = 0$  pour toute solution réalisable.

Cependant, si l'on se restreint à des classes particulières de matroïdes qui ne contiennent pas ces instances conflictuelles, nous pouvons trouver des compromis non triviaux. Par exemple, on peut s'intéresser à la classe des matroïdes *coloriés-simples* (voir la définition 1.17, page 39) qui généralise les matroïdes simples (définition 1.14, page 26). Les matroïdes des exemples 6.2 et 6.3 ne sont pas coloriés-simples. Plus généralement, un matroïde simple est colorié-simple et lorsque chaque couleur apparaît une fois, ces deux notions coïncident.

Comme nous le montrerons dans la prochaine section, l'exclusion des matroïdes qui ne sont pas coloriés-simples fait que l'existence d'une approximation non triviale est possible. Nous supposons également que le rang  $r(\mathcal{M})$  du matroïde  $\mathcal{M}$  est au moins 2.

## 6.2 Un algorithme général glouton

Nous proposons d'analyser une extension simple du GLOUTON (algorithme 1.1, page 32) qui construit une solution de compromis. Cette extension, appelée ALT-GLOUTON, simule un processus simple et naturel pour la construction d'un compromis. ALT-GLOUTON est décrit dans l'algorithme 6.1.

Soit *l'agent poids* celui qui veut maximiser  $w(F)$  et *l'agent couleur* celui qui veut maximiser  $g(F)$ . Au début,  $F = \emptyset$  et les agents ajoutent, de façon alternée, un élément  $e$  à  $F$  tel que  $F + e \in \mathcal{F}$  jusqu'à ce que  $F$  devienne une base. Si c'est le tour de l'agent poids, alors ce dernier sélectionne un élément  $e$  qui maximise  $w(F+e)$ . De manière symétrique, l'agent couleur choisit un élément  $e$  qui maximise  $g(F+e)$  durant son tour. Nous supposons que l'agent poids commence en premier.

**Théorème 6.1** ALT-GLOUTON est  $(\frac{1}{2}, \frac{1}{4})$ -approché pour des matroïdes coloriés-simples.

**Preuve.** Soit  $B = \{e_1, \dots, e_r\}$  la base retournée par ALT-GLOUTON. Nous nous concentrons d'abord sur le rapport d'approximation  $1/2$  pour l'agent poids. Soit

**Algorithme 6.1 : ALT-GLOUTON**

**Données :** un matroïde  $\mathcal{M} = (X, \mathcal{F})$ , une fonction poids  $w : X \rightarrow \mathbb{R}^+$ , une fonction couleur  $\mathcal{L} : X \rightarrow \{\ell_1, \dots, \ell_p\}$ , un fonction gain des couleurs  $g : \{\ell_1, \dots, \ell_p\} \rightarrow \mathbb{R}^+$

**Résultat :** une base de  $\mathcal{M}$

```

1  $F \leftarrow \emptyset$ 
2  $R \leftarrow \{e \in X : F + e \in \mathcal{F}\}$ 
3  $i \leftarrow 1$ 
4 tant que  $R \neq \emptyset$  faire
5   | si  $i = 1$  alors
6   |   | choisir  $e \in R$  de poids  $w$  maximum
7   | sinon
8   |   | si  $\mathcal{L}(R) \setminus \mathcal{L}(F) \neq \emptyset$  alors
9   |   |   | choisir  $e \in R$  tel que  $\mathcal{L}(e) \notin F$  et  $g(\mathcal{L}(e))$  est maximum
10  |   | sinon
11  |   |   | choisir  $e \in R$ 
12  |    $F \leftarrow F + e$ 
13  |    $R \leftarrow \{e \in X \setminus F : F + e \in \mathcal{F}\}$ 
14  |    $i \leftarrow 3 - i$ 
15 retourner  $F$ 

```

$B^w = \{e_1^w, \dots, e_r^w\}$  une base de  $\mathcal{M}$  de poids maximum vérifiant  $w(e_1^w) \geq w(e_2^w) \geq \dots \geq w(e_r^w)$ . Puisque l'agent poids commence à choisir le premier, alors chaque élément de la solution  $B$  avec un indice impair  $i$  est inséré par l'agent poids qui veut maximiser le poids total.

Soit  $i \leq r$  un entier impair. Notons  $B_i^w = \{e_1^w, \dots, e_i^w\}$  l'ensemble contenant les  $i$  premiers éléments de  $B^w$ . Considérons la solution courante  $F = \{e_1, \dots, e_{i-1}\}$  de ALT-GLOUTON juste avant d'insérer  $e_i$ . En appliquant l'axiome (iii) des matroïdes (voir la définition 1.12, page 24) et puisque  $|B_i^w| > |F|$ , alors il existe un élément  $\hat{e} \in B_i^w \setminus F$  tel que  $F + \hat{e} \in \mathcal{F}$ . En utilisant  $w(e_1^w) \geq \dots \geq w(e_i^w)$ , on déduit que  $w(\hat{e}) \geq w(e_i^w)$ . Puisque l'agent poids sélectionne l'élément  $e_i$  qui maximise  $w(F + e_i)$  et comme la fonction d'utilité de cet agent est additive, alors  $w(e_i) \geq w(\hat{e}) \geq w(e_i^w)$  pour tout indice impair  $i \leq r$ . En combinant cette inégalité avec toutes les valeurs de  $i$ , on obtient

$$w(B) = w\left(\bigcup_{i=1}^r \{e_i\}\right) \geq w\left(\bigcup_{i=1, \text{ impair}}^r \{e_i\}\right) \geq w\left(\bigcup_{i=1, \text{ impair}}^r \{e_i^w\}\right) \quad (6.1)$$

où la positivité du poids d'un élément est utilisée.

Observons que  $w(e_j^w) \geq \frac{1}{2}(w(e_j^w) + w(e_{j+1}^w))$  car  $w(e_j^w) \geq w(e_{j+1}^w)$  pour  $j = 1, \dots, r-1$ . En utilisant (6.1), on obtient

$$w(B) \geq w\left(\bigcup_{i=1, \text{ impair}}^r \{e_i^w\}\right) \geq \frac{1}{2}w\left(\bigcup_{i=1}^r \{e_i^w\}\right) = \frac{1}{2}w(B^w)$$

D'où, ALT-GLOUTON est  $(1/2, \cdot)$ -approché.

Considérons maintenant l'agent couleur qui ajoute un élément à la solution à chaque étape paire. Soit  $B^g$  une base avec un gain  $g$  maximum. Soit  $E$  l'ensemble des éléments de  $B^g$  dont la couleur n'apparaît pas dans la solution retournée par ALT-GLOUTON, c'est-à-dire  $E = \{e \in B^g : \mathcal{L}(e) \notin \mathcal{L}(B)\}$ . Nous supposons que les  $\nu = |\mathcal{L}(E)|$  premiers éléments de  $E$  ont des couleurs distinctes et qu'ils sont classés par ordre décroissant du gain :  $g(e_1^g) \geq g(e_2^g) \geq \dots \geq g(e_\nu^g)$  où  $E = \{e_1^g, \dots, e_\nu^g, \dots, e_{|E|}^g\}$ .  $E$  appartient à  $\mathcal{F}$  car  $E \subseteq B^g$  et  $B^g \in \mathcal{F}$ . Notons que  $g(E) = g(\{e_1^g, \dots, e_\nu^g\}) = \sum_{i=1}^{\nu} g(e_i^g)$ .

Supposons que durant une étape paire de ALT-GLOUTON, l'agent couleur ne peut pas ajouter un élément avec une nouvelle couleur à la solution courante  $F$ . Nous savons que  $F \in \mathcal{F}$  car  $F \subseteq B$ . Si  $|F| < |E|$  alors grâce à l'axiome (iii) des matroïdes, on peut ajouter un élément avec une nouvelle couleur à  $F$ , contradiction. On déduit que  $|F| \geq |E|$ . Ainsi  $F$  a au moins  $|\mathcal{L}(E)| = \nu$  éléments.

Soit  $i \leq \nu$  un entier pair. Soit  $B_i^g = \{e_1^g, \dots, e_i^g\}$  l'ensemble contenant les  $i$  premiers éléments de  $B^g$  tels que  $e_1^g, \dots, e_i^g$  ont des couleurs distinctes (ceci est possible car  $\nu = |\mathcal{L}(E)| \leq |\mathcal{L}(B^g)|$ ) et  $g(e_1^g) \geq \dots \geq g(e_i^g)$ . Considérons la solution courante  $F = \{e_1, \dots, e_{i-1}\}$  de ALT-GLOUTON juste avant d'insérer  $e_i$ . En appliquant l'axiome (iii) des matroïdes et puisque  $|B_i^g| > |F|$ , il existe un élément  $\tilde{e} \in B_i^g \setminus F$  tel que  $F + \tilde{e} \in \mathcal{F}$ . En utilisant  $g(e_1^g) \geq \dots \geq g(e_i^g)$ , nous savons que  $g(\tilde{e}) \geq g(e_i^g)$ . L'agent couleur sélectionne l'élément  $e_i$  avec une nouvelle couleur et qui maximise  $g(F + e_i)$ , donc  $g(e_i) \geq g(\tilde{e}) \geq g(e_i^g)$ , pour tout indice pair  $i \leq \nu$ . Puisque  $\{e_1, \dots, e_\nu\} \subseteq B$ , alors

$$g(B) = g\left(\bigcup_{i=1}^{\nu} \{e_i\}\right) \geq g\left(\bigcup_{i=2, \text{pair}}^{\nu} \{e_i\}\right) \geq g\left(\bigcup_{i=2, \text{pair}}^{\nu} \{e_i^g\}\right) \quad (6.2)$$

Les  $\nu$  premiers éléments de  $E$  devant être triés par gain décroissant, nous savons que  $g(e_j^g) \geq \frac{1}{2}(g(e_j^g) + g(e_{j+1}^g))$  pour tout  $j = 1, \dots, \nu - 1$ . En utilisant l'inégalité (6.2), on obtient  $g(B) \geq g\left(\bigcup_{i=2, \text{pair}}^{\nu} \{e_i^g\}\right) \geq \frac{1}{2}g\left(\bigcup_{i=2}^{\nu} \{e_i^g\}\right)$ . Puisque  $g\left(\bigcup_{i=1}^{\nu} \{e_i^g\}\right) = g(E)$ , alors

$$2g(B) \geq g(E) - g(e_1^g) \quad (6.3)$$

Si  $\mathcal{L}(e_1) = \mathcal{L}(e_1^g)$  alors  $g(B) \geq g(e_1^g)$ . Sinon,  $\{e_1, e_1^g\} \in \mathcal{F}$  car  $r(\mathcal{M}) \geq 2$  et  $\mathcal{M}$  est colorié-simple (voir la définition 1.17, page 39). De plus,  $g(e_2) \geq g(e_1^g)$  en raison de l'axiome (iii) des matroïdes et l'agent couleur sélectionne un élément qui maximise  $g$ . Finalement, dans les deux cas, on obtient

$$g(B) \geq g(e_1^g) \quad (6.4)$$

En utilisant les inégalités (6.3) et (6.4), on obtient

$$3g(B) \geq g(E) \quad (6.5)$$

D'après la définition de  $E$ ,  $\mathcal{L}(B)$  contient les couleurs de  $\mathcal{L}(B^g) \setminus \mathcal{L}(E)$ . Alors  $g(B) \geq g(B^g) - g(E)$  qu'on ajoute à l'inégalité (6.5) et on obtient  $4g(B) \geq g(B^g)$ . Ainsi, ALT-GLOUTON est  $(\cdot, 1/4)$ -approché si, durant une étape paire de ALT-GLOUTON, l'agent couleur ne peut pas ajouter un élément avec une nouvelle couleur à la solution courante  $F$ .

Si l'agent couleur avait été capable d'ajouter un élément avec une nouvelle couleur à chaque étape paire alors la solution aurait été meilleure de son point de vue. En effet, renommons les  $\xi = |\mathcal{L}(B^g)|$  premiers éléments de  $B^g$  tels que  $\mathcal{L}(\{e_1^g, \dots, e_\xi^g\}) = \mathcal{L}(B^g)$  et  $g(e_1^g) \geq g(e_2^g) \geq \dots \geq g(e_\xi^g)$ . Notons que  $g(\bigcup_{i=1}^\xi \{e_i^g\}) = g(B^g)$ . En utilisant l'axiome (iii) des matroïdes, on a  $g(e_i) \geq g(e_i^g)$  pour  $i \leq \xi$  et pair. On déduit que

$$g(B) \geq g\left(\bigcup_{i=2, \text{pair}}^\xi \{e_i\}\right) \geq g\left(\bigcup_{i=2, \text{pair}}^\xi \{e_i^g\}\right) \quad (6.6)$$

Puisque les  $\xi$  premiers éléments de  $B^g$  sont triés par gain décroissant, alors

$$2g\left(\bigcup_{i=2, \text{pair}}^\xi \{e_i^g\}\right) \geq g\left(\bigcup_{i=1}^\xi \{e_i^g\}\right) - g(e_1^g) \quad (6.7)$$

En utilisant (6.6) et (6.7), on obtient  $2g(B) \geq g(B^g) - g(e_1^g)$ . Puisque l'inégalité (6.4) est valide, alors  $3g(B) \geq g(B^g)$  qui est meilleur que  $4g(B) \geq g(B^g)$ . ■

Le prochain exemple montre que l'analyse de ALT-GLOUTON est au plus près (*tight*).

**Exemple 6.4** Considérons le matroïde graphique du graphe  $G = (V, E)$  représenté dans la figure 6.1.

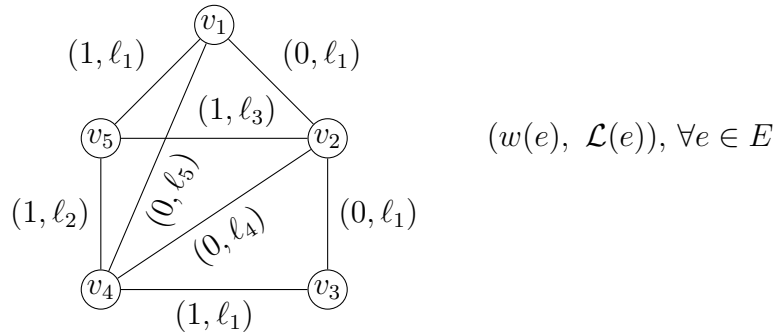


FIGURE 6.1 – Instance critique de ALT-GLOUTON pour des matroïdes coloriés-simples.

Observons que le matroïde associé à ce graphe est colorié-simple car il ne contient pas de couple d'arêtes qui ne soit pas une forêt (voir la définition 1.17,

page 39). Nous supposons que  $g(\ell_2) = 0$  et  $g(\ell_i) = 1$  pour tout  $i \neq 2$ . L'arbre  $\{(v_1, v_5), (v_2, v_5), (v_3, v_4), (v_4, v_5)\}$  a un poids de 4 alors que l'arbre  $\{(v_1, v_4), (v_2, v_3), (v_2, v_4), (v_2, v_5)\}$  utilise les couleurs  $\{\ell_1, \ell_3, \ell_4, \ell_5\}$ , donc son gain est 4. L'algorithme ALT-GLOUTON peut retourner l'arbre contenant  $(v_4, v_5), (v_1, v_2), (v_1, v_5)$  et  $(v_2, v_3)$  (les arêtes sont données dans l'ordre de leur ajout dans la solution). Cet arbre a un poids de 2 et son gain est 1.

Si on applique ALT-GLOUTON en inversant l'ordre des agents (les étapes paires sont pour l'agent poids et les étapes impaires sont pour l'agent couleur), alors on peut montrer que ALT-GLOUTON est  $(1/3, 1/3)$ -approché pour des matroïdes simples.

**Théorème 6.2** *En inversant l'ordre de choix des agents, ALT-GLOUTON est  $(\frac{1}{3}, \frac{1}{3})$ -approché pour des matroïdes simples.*

**Preuve.** Nous gardons les mêmes notations que celles du théorème 6.1. Nous nous concentrons d'abord sur le rapport d'approximation  $1/3$  pour l'agent couleur.

Supposons que durant une étape impaire de ALT-GLOUTON, l'agent couleur ne peut pas ajouter d'élément avec une nouvelle couleur à la solution courante  $F$ . Nous savons que  $|F| \geq |E|$  et  $F$  a au moins  $\nu$  éléments (voir la preuve du théorème 6.1, page 146). En utilisant l'axiome (iii) des matroïdes, nous savons que  $g(e_i) \geq g(e_i^g)$  est valide pour chaque  $i$  impair compris entre 1 et  $\nu$ . Il s'ensuit que  $\sum_{i=1, \text{impair}}^{\nu} g(e_i) \geq \sum_{i=1, \text{impair}}^{\nu} g(e_i^g)$ . Les  $\nu$  premiers éléments de  $E$  devant être triés par gain décroissant, on a

$$2 \sum_{i=1, \text{impair}}^{\nu} g(e_i^g) \geq \sum_{i=1}^{\nu} g(e_i^g) = g(E)$$

Puisque  $\{e_1, \dots, e_{\nu}\} \subseteq B$ , alors

$$2g(B) \geq 2 \sum_{i=1, \text{impair}}^{\nu} g(e_i) \geq 2 \sum_{i=1, \text{impair}}^{\nu} g(e_i) \geq \sum_{i=1}^{\nu} g(e_i^g) = g(E)$$

Rappelons que  $g(B) \geq g(B^g) - g(E)$ , donc en l'additionnant à la dernière inégalité, on obtient  $3g(B) \geq g(B^g)$ .

Supposons maintenant que l'agent couleur ajoute un élément avec une nouvelle couleur durant chaque étape impaire. En utilisant l'axiome (iii) des matroïdes,  $g(e_i) \geq g(e_i^g)$  pour  $i \leq \xi$  et impair où  $\xi = |\mathcal{L}(B^g)|$ . Alors  $g(B) \geq \sum_{i=1, \text{impair}}^{\xi} g(e_i) \geq \sum_{i=1, \text{impair}}^{\xi} g(e_i^g)$ . Puisque les  $\xi$  premiers éléments de  $B^g$  sont triés par gain décroissant, alors  $2 \sum_{i=1, \text{impair}}^{\xi} g(e_i^g) \geq \sum_{i=1}^{\xi} g(e_i^g)$ . En combinant ces deux inégalités, on obtient  $2g(B) \geq g(B^g)$  qui est meilleur que  $3g(B) \geq g(B^g)$ .

Considérons maintenant l'agent poids. Chaque élément de  $B$  avec un indice pair est inséré par l'agent poids qui maximise  $w$ . Soit  $i \leq r$  un entier pair. En utilisant les mêmes arguments que dans le théorème 6.1,  $w(e_i) \geq w(e_i^w)$  pour chaque  $i \leq r$  pair, donc  $\sum_{i=1}^r w(e_i) \geq \sum_{i=2, \text{ pair}}^r w(e_i) \geq \sum_{i=2, \text{ pair}}^r w(e_i^w)$ . On observe que  $w(e_i^w) \geq \frac{1}{2} (w(e_i^w) + w(e_{i+1}^w))$  car  $w(e_i^w) \geq w(e_{i+1}^w)$  pour chaque  $i \leq r-1$  pair. Ainsi,

$$2w(B) = 2 \sum_{i=1}^r w(e_i) \geq \sum_{i=2}^r w(e_i^w) = w(B^w) - w(e_1^w) \quad (6.8)$$

Si  $w(e_1) = w(e_1^w)$  alors  $w(B) \geq w(e_1^w)$ , sinon  $\{e_1, e_1^w\} \in \mathcal{F}$  car le matroïde est simple (voir la définition 1.14, page 26). Soit  $e_2$  le second élément inséré dans  $B$  (par hypothèse  $r(\mathcal{M}) \geq 2$ ). Puisque l'agent poids sélectionne  $e_2$  afin de maximiser  $w(\{e_1, e_2\})$ , alors  $w(e_2) \geq w(e_1^w)$ . Dans les deux cas,  $w(B) \geq w(e_1^w)$ . En l'additionnant à l'inégalité (6.8), on obtient finalement  $3w(B) \geq w(B^w)$ . ■

Le résultat précédent est au plus près (*tight*) comme on peut le voir dans l'exemple 6.5.

**Exemple 6.5** Considérons le matroïde graphique du graphe simple décrit dans la figure 6.2.

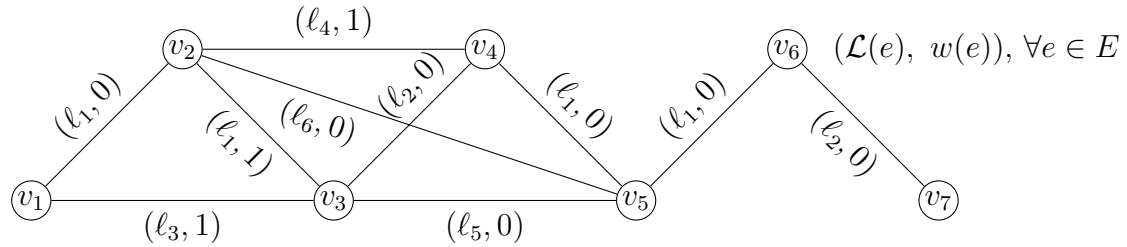


FIGURE 6.2 – Instance critique de ALT-GLOUTON pour des matroïdes simples.

Le matroïde associé à un graphe simple est toujours simple car dans un graphe simple, chaque arête ou couple d'arête est une forêt (voir la définition 1.14, page 26).

Le gain de chaque couleur est 1. L'arbre  $T_g = \{(v_1, v_3), (v_3, v_5), (v_2, v_4), (v_2, v_5), (v_5, v_6), (v_6, v_7)\}$  a un gain maximum de 6 et l'arbre  $T_w = \{(v_1, v_3), (v_2, v_3), (v_2, v_4), (v_4, v_5), (v_5, v_6), (v_6, v_7)\}$  a un poids maximum de 3. ALT-GLOUTON commence avec l'agent couleur suivi par l'agent poids. L'algorithme peut prendre les arêtes dans l'ordre suivant :  $(v_1, v_2)$ ,  $(v_2, v_3)$ ,  $(v_3, v_4)$ ,  $(v_4, v_5)$ ,  $(v_5, v_6)$  et  $(v_6, v_7)$ . Le gain est donc 2 et le poids est 1.



### 6.3 Utilités additives

Nous supposons dans cette section que chaque élément de  $X$  a une couleur distincte, c'est-à-dire  $|\mathcal{L}(X)| = |X|$ . L'utilité du second agent consiste alors à additionner le gain de tous les éléments d'un ensemble indépendant, donc  $g$  devient additive.

Nous retrouvons ici le problème traité dans les chapitres précédents pour deux agents. Rappelons que ce problème n'admet pas d'approximation à rapport constant (voir la section 6.1.2, page 145 et l'exemple 2.8, page 57). Pour pallier cette situation, nous avons proposé dans les chapitres 4 et 5 des approximations dépendant des paramètres  $\alpha_i$ ,  $i = 1, \dots, n$ . Dans ce chapitre, nous nous focalisons sur des classes particulières de matroïdes et nous proposons, pour ces classes, des approximations à rapports constants.

Nous proposons une généralisation de ALT-GLOUTON où les éléments de la base résultante sont choisis en ajoutant, de façon alternée,  $x$  éléments qui maximisent  $w$  et  $y$  éléments qui maximisent  $g$  pour des entiers  $x, y \geq 1$ . Si  $x = 0$  ou  $y = 0$  (non tous les deux nuls), cela revient à appliquer le GLOUTON (algorithme 1.1, page 32) respectivement pour  $(\mathcal{M}, g, \max)$  ou  $(\mathcal{M}, w, \max)$ . Nous supposons que  $r(\mathcal{M}) \geq x + 1$  car autrement il n'y aurait pas de garantie. Sans perte de généralité, nous supposons que  $r(\mathcal{M}) = p(x + y)$  pour un entier  $p \geq 1$ . Si cela n'est pas vérifié (c'est-à-dire si  $r(\mathcal{M}) > p(x + y)$ ), alors on complète la solution partielle  $B$  ( $B$  est ici un ensemble indépendant de  $p(x + y)$  éléments) avec un ensemble d'éléments  $R$  de sorte que  $B \cup R$  soit une base. Puisque les utilités sont positives, alors  $w(B \cup R) \geq w(B)$  et  $g(B \cup R) \geq g(B)$ . Cet algorithme est appelé ALT-GLOUTON GÉNÉRALISÉ.

Nous obtenons le résultat suivant en utilisant la notion de matroïde  $k$ -simple (définition 1.15, page 26).

**Théorème 6.3** ALT-GLOUTON GÉNÉRALISÉ est  $\left(\frac{x}{x+y}, \frac{y}{x+y+xy}\right)$ -approché pour tout matroïde  $(x + 1)$ -simple  $\mathcal{M}$  vérifiant  $r(\mathcal{M}) = p(x + y)$  pour tous les entiers  $p, x, y \geq 1$ , et lorsque chaque couleur apparaît au plus une fois.

**Preuve.** Soit  $B = \cup_{i=1..p} X_i \cup Y_i$ ,  $|X_i| = x$ ,  $|Y_i| = y$  l'ensemble indépendant retourné par ALT-GLOUTON GÉNÉRALISÉ en respectant l'ordre d'insertion de ces ensembles dans l'algorithme. Les éléments de  $X_i$  maximisent  $w$  et ceux de  $Y_i$  maximisent  $g$  pour  $i = 1..p$ . Nous voulons montrer que  $w(B) \geq \frac{x}{x+y}w(B^w)$  et  $g(B) \geq \frac{y}{x+y+xy}g(B^g)$  où  $B^w$  et  $B^g$  sont des bases maximales respectivement pour  $w$  et  $g$ .

Nous nous concentrons d'abord sur le rapport d'approximation  $x/(x + y)$  pour le poids  $w$ . Supposons que les éléments de  $B^w$  sont triés par ordre décroissant de  $w$

et  $B^w = \bigcup_{i=1..p} (X_i^w \cup Y_i^w)$  où  $|X_i^w| = x$  et  $|Y_i^w| = y$ . Soit  $i$  un entier dans  $\{1, \dots, p\}$ . La solution courante juste avant que  $X_i$  soit inséré est  $F = \bigcup_{j=1..i-1} X_j \cup Y_j$ . Soit  $B_i^w = \bigcup_{j=1..i-1} (X_j^w \cup Y_j^w) \cup X_i^w$ . En utilisant la propriété (iii') des matroïdes (voir la définition 1.12, page 24), et puisque  $|B_i^w| > |F|$ , on déduit qu'il existe  $A \subseteq B_i^w \setminus F$  avec  $|A| = |B_i^w| - |F| = x$  tel que  $F \cup A \in \mathcal{F}$ . En utilisant  $w(X_1^w) \geq w(X_2^w) \geq \dots \geq w(X_p^w)$ , on a  $w(A) \geq w(X_i^w)$ . Puisque l'agent poids sélectionne l'ensemble  $X_i$  de taille  $x$  qui maximise  $w(F \cup X_i)$  et comme la fonction d'utilité de cet agent est additive, alors  $w(X_i) \geq w(A) \geq w(X_i^w)$  pour tout  $i \in \{1, \dots, p\}$ . Ainsi,

$$w(B) = \sum_{i=1}^p (w(X_i) + w(Y_i)) \geq \sum_{i=1}^p w(X_i) \geq \sum_{i=1}^p w(X_i^w) \quad (6.9)$$

Puisque les éléments de  $B^w$  sont triés par ordre décroissant de  $w$ , et  $x, y \geq 1$ , alors  $yw(X_i^w) \geq xw(Y_i^w)$ . En ajoutant  $xw(X_i^w)$  des deux côtés de l'inégalité, on obtient  $(x+y)w(X_i^w) \geq x(w(X_i^w) + w(Y_i^w))$ . L'inégalité (6.9) devient

$$w(B) \geq \sum_{i=1}^p w(X_i^w) \geq \frac{x}{x+y} \sum_{i=1}^p (w(X_i^w) + w(Y_i^w)) = \frac{x}{x+y} w(B^w)$$

D'où, ALT-GLOUTON GÉNÉRALISÉ est  $(\frac{x}{x+y}, \cdot)$ -approché.

Considérons maintenant l'agent couleur. Supposons que  $B^g = \bigcup_{i=1..p} X_i^g \cup Y_i^g$  où  $|X_i^g| = x$  et  $|Y_i^g| = y$  et tels que les éléments de  $B^g$  sont triés par ordre décroissant de  $g$ . Puisque  $g$  est également additive, on obtient (voir plus tôt),

$$g(B) = \sum_{i=1}^p (g(X_i) + g(Y_i)) \geq \sum_{i=1}^p g(Y_i) \geq \sum_{i=1}^p g(Y_i^g) \quad (6.10)$$

Puisque les éléments de  $B^g$  sont triés par ordre décroissant de  $g$ , et  $x, y \geq 1$ , alors  $xg(Y_i^g) \geq yg(X_{i+1}^g)$  pour  $i < p$ . En ajoutant  $yg(Y_i^g)$  des deux côtés de la dernière inégalité, on obtient  $(x+y)g(Y_i^g) \geq y(g(Y_i^g) + g(X_{i+1}^g))$  pour  $i < p$ . L'inégalité (6.10) devient

$$g(B) \geq \sum_{i=1}^p g(Y_i^g) \geq \frac{y}{x+y} \sum_{i=1}^{p-1} (g(Y_i^g) + g(X_{i+1}^g)) + g(Y_p^g) \geq \frac{y}{x+y} (g(B^g) - g(X_1^g))$$

car  $y/(x+y) < 1$ . Donc,

$$\frac{x+y}{y} g(B) \geq g(B^g) - g(X_1^g) \quad (6.11)$$

Soit  $e_1^g$  l'élément de  $X_1^g$  de gain maximum. Puisque  $B^g$  est trié par ordre décroissant de  $g$ , on obtient  $g(X_1^g) \leq xg(e_1^g)$ . De (6.11), on a

$$\frac{x+y}{y} g(B) \geq g(B^g) - xg(e_1^g) \quad (6.12)$$

S'il existe  $e \in X_1$  tel que  $g(e) = g(e_1^g)$  alors  $g(B) \geq g(e_1^g)$ , sinon  $X_1 \cup \{e_1^g\} \in \mathcal{F}$  car le matroïde est  $(x + 1)$ -simple (voir la définition 1.15 page 26). Soit  $e_1$  le premier élément inséré dans  $B$  après  $X_1$  (par hypothèse,  $r(\mathcal{M}) \geq x + 1$ ). Puisque le second agent (l'agent couleur) sélectionne  $e_1$  dans le but de maximiser  $g(X_1 \cup \{e_1\})$ , alors  $g(e_1) \geq g(e_1^g)$ . Dans tous les cas, on obtient  $g(B) \geq g(e_1^g)$ , donc

$$xg(B) \geq xg(e_1^g) \tag{6.13}$$

En additionnant (6.12) et (6.13), on obtient  $\frac{x+y+xy}{y}g(B) \geq g(B^g)$ . D'où, ALT-GLOUTON GÉNÉRALISÉ est  $(\frac{x}{x+y}, \frac{y}{x+y+xy})$ -approché. ■

Lorsque  $x = y = 1$ , cela revient à appliquer ALT-GLOUTON pour obtenir une  $(1/2, 1/3)$ -approximation pour des matroïdes simples. Cette approximation est au plus près (*tight*) comme on peut le voir dans l'exemple 6.6.

**Exemple 6.6** Nous présentons une instance du matroïde graphique définie sur un graphe connexe simple  $G = (V, E)$  représenté dans la figure 6.3.

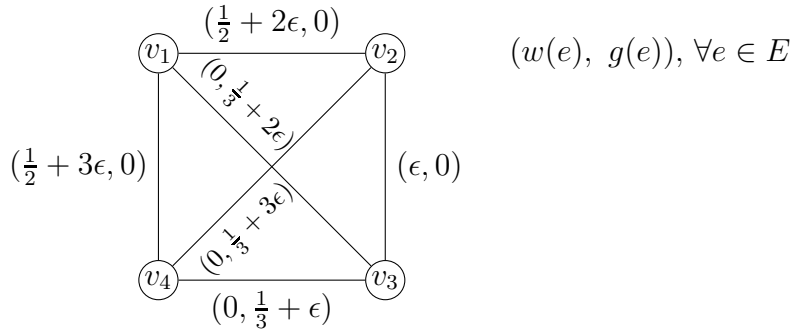


FIGURE 6.3 – Instance critique de ALT-GLOUTON GÉNÉRALISÉ pour des matroïdes simples.

Nous avons  $B^w = \{(v_1, v_2), (v_1, v_4), (v_2, v_3)\}$  et  $B^g = \{(v_1, v_3), (v_2, v_4), (v_3, v_4)\}$  avec  $w(B^w) = g(B^g) = 1 + 6\epsilon$  où  $B^w$  et  $B^g$  correspondent à des arbres couvrants maximums respectivement pour  $(G, w)$  et  $(G, g)$  (en appliquant le GLOUTON ou l'algorithme de Kruskal).

Lorsque  $x = y = 1$ , alors ALT-GLOUTON GÉNÉRALISÉ produit l'arbre couvrant  $B = \{(v_1, v_4), (v_2, v_4), (v_2, v_3)\}$  où  $w(B) = 1/2 + 4\epsilon$  et  $g(B) = 1/3 + 3\epsilon$ . Ainsi,  $w(B)/w(B^w) \rightarrow 1/2$  et  $g(B)/g(B^g) \rightarrow 1/3$  lorsque  $\epsilon \rightarrow 0$ . De plus, cette approximation est la meilleure qu'on puisse avoir car il n'existe pas d'arbre  $B'$  de  $G$  tel que  $w(B') \geq w(B)$ ,  $g(B') \geq g(B)$  et  $(w(B'), g(B')) \neq (w(B), g(B))$ .

## 6.4 Complexité générale et une solution particulière

Le théorème 6.1 (page 146) est un résultat constructif sur le fait que chaque instance coloriée-simple du modèle admet une solution  $(1/2, 1/4)$ -approchée. Plus généralement, étant donnée une instance  $((X, \mathcal{F}), w, \mathcal{L}, g)$  et deux bornes  $k_w$  et  $k_g$ , quelle est la complexité du problème de décision suivant ?

$\Pi$  : Existe-t-il  $F \in \mathcal{F}$  tel que  $w(F) \geq k_w$  et  $g(F) \geq k_g$  ?

$\Pi$  généralise le problème de décision associé à l'arbre couvrant de poids minimum avec une contrainte additionnelle qui est NP-complet (Aggarwal *et al.*, 1982), voir aussi Ehrgott (1996) pour les matroïdes. En considérant le matroïde graphique où chaque arête a une couleur distincte, le problème traité dans ce chapitre est la version maximisation du problème de l'arbre couvrant de poids minimum avec une contrainte additionnelle.

Bien que  $\Pi$  soit NP-complet en général, le résultat suivant (théorème 6.4, page 156) montre qu'une solution Pareto-optimale lexicographique particulière peut être obtenue en temps polynomial. Parmi l'ensemble des solutions optimales pour  $w$ , soit  $F_{w,g}^*$  celle qui maximise  $g$ . Soit  $\mathcal{B}$  l'ensemble des bases de  $\mathcal{M}$  et  $\mathcal{B}_w \subseteq \mathcal{B}$  l'ensemble des bases qui sont optimales pour  $w : X \rightarrow \mathbb{R}^+$ . Alors  $F_{w,g}^*$  consiste à résoudre  $\max_{B \in \mathcal{B}_w} g(B)$ . En d'autres termes,  $F_{w,g}^*$  est une base de poids maximum qui maximise le gain des couleurs utilisées. Par exemple, si le matroïde est graphique dans un graphe connexe pondéré  $G = (V, E; w)$ , alors nous recherchons un arbre couvrant  $T$  qui maximise  $\sum_{\ell \in \mathcal{L}(T)} g(\ell)$  parmi les arbres couvrants de poids maximum.

Nous observons les faits suivants :

- Si  $w(e) = 1$  pour toute arête  $e \in E$  et  $g(\ell) = 1$  pour toute couleur  $\ell$ , alors on recherche un arbre couvrant avec un nombre maximum de couleurs, c'est-à-dire trouver  $\max_{T \in \mathcal{T}} |\mathcal{L}(T)|$  où  $\mathcal{T}$  est l'ensemble des arbres couvrants. Ce problème est montré polynomial par Broersma et Li (1997). Plus généralement, comme le mentionne Maffioli *et al.* (2007), trouver une base  $B$  d'un matroïde  $\mathcal{M} = (X, \mathcal{F})$  qui maximise le prix chromatique  $\sum_{\ell \in \mathcal{L}(B)} g(\ell)$  peut être facilement réduit à un problème d'intersection de matroïdes. Ainsi, ce problème est résoluble en temps polynomial et il correspond à trouver  $F_{w,g}^*$  dans le cas particulier où  $w(x) = 1$  pour tout élément  $x \in X$ . Nous proposerons par la suite de résoudre ce problème dans le cas général.
- La complexité d'obtention d'une base d'un matroïde  $\mathcal{M} = (X, \mathcal{F})$  de poids minimum qui maximise le prix chromatique (résoudre  $\max_{B \in \mathcal{B}_w^-} g(B)$  où  $\mathcal{B}_w^-$  est l'ensemble des bases de poids minimum) est polynomialement équivalent

à trouver  $F_{w,g}^*$  en posant  $w'(x) = w_{\max} - w(x)$ , pour tout  $x \in X$  où  $w_{\max} = \max_{x \in X} w(x)$ .

- Par ailleurs, le problème de la base de poids maximum d'un matroïde qui minimise le prix chromatique, c'est-à-dire  $\min_{B \in \mathcal{B}_w} g(B)$  est plus difficile (Maffioli *et al.*, 2007, problème 3, page 1962). Ce problème est  $\ln r(\mathcal{M})$ -approché (Maffioli *et al.*, 2007) et cette approximation est au plus près (*tight*) et ne peut être améliorée.

Notons que  $F_{w,g}^*$  est une solution Pareto-optimale lexicographique.

**Théorème 6.4**  $F_{w,g}^*$  peut être obtenue en temps polynomial.

**Proof.** Soit  $\mathcal{B}_w \subseteq \mathcal{B}$  l'ensemble des bases maximums pour  $w$  dans le matroïde  $\mathcal{M} = (X, \mathcal{F})$ . La collection  $\mathcal{B}_w$  vérifie la propriété 1.1 page 37. En utilisant le théorème 1.1 (page 25), on observe que  $\mathcal{B}_w$  est la collection des bases du matroïde  $\mathcal{M}_w = (X, \mathcal{F}_w)$  où  $\mathcal{F}_w = \{F \subseteq X : \exists B \in \mathcal{B}_w \text{ telle que } F \subseteq B\}$ .

Soit  $\mathcal{M}' = (X, \mathcal{F}')$  un matroïde de partition (voir page 30) où  $F \in \mathcal{F}'$  si et seulement si  $F \subseteq X$  et  $|\mathcal{L}(F)| = |F|$ . Nous voulons trouver  $\hat{F} \in \mathcal{F}_w \cap \mathcal{F}'$  tel que  $g(\hat{F})$  est maximum. Puisque  $\mathcal{M}_w$  et  $\mathcal{M}'$  sont deux matroïdes définis sur le même ensemble d'éléments  $X$ , alors l'intersection de  $\mathcal{M}_w$  et  $\mathcal{M}'$  où  $g$  est maximisé peut être résolue en temps polynomial (Frank, 1981). Puis, il suffit de compléter  $\hat{F}$  en une base  $\hat{B}$  de façon gloutonne avec des éléments de  $X$  triés par poids décroissant (comme pour l'algorithme GLOUTON). On conclut que  $\hat{B} = F_{w,g}^*$  car  $g(\hat{B}) = g(\hat{F})$ ,  $\hat{F}$  maximise  $g$  et contient un nombre maximum de couleurs. ■

La complexité de l'algorithme qui trouve  $F_{w,g}^*$  dépend de l'algorithme d'intersection de deux matroïdes qui est en  $O(|X^3|\theta)$ .

Une question ouverte est de connaître la complexité d'obtention de l'autre solution Pareto-optimale  $F_{g,w}^*$  qui correspond à  $\max_{B \in \mathcal{B}_g} w(B)$  où  $\mathcal{B}_g$  est l'ensemble de toutes les bases maximums pour  $g$ . En d'autres termes,  $F_{g,w}^*$  est une base  $B$  de  $\mathcal{M}$  avec un prix chromatique maximum qui maximise le poids  $w(B) = \sum_{x \in B} w(x)$ . Nous n'avons pas pu résoudre ce problème mais nous donnons quelques éléments de réponse dans la section 6.5.2 (page 161).

## 6.5 Cas des couleurs non-pondérées

Dans cette section, nous considérons le cas particulier des *couleurs non pondérées*, où  $g(\ell) = 1$  pour chaque couleur  $\ell$ . La définition  $g(F) = |\mathcal{L}(F)|$  est valide dans ce cas. Donc l'objectif de l'agent couleur est de maximiser le nombre de couleurs distinctes. Comme mentionné précédemment, trouver une base  $B^c$  qui

maximise  $|\mathcal{L}(B^\mathcal{L})|$  peut être obtenu en temps polynomial. Dans ce qui suit,  $L$  représente  $|\mathcal{L}(B^\mathcal{L})|$ . Dans cette section, nous ne supposons pas que le matroïde est colorié-simple.

### 6.5.1 Un algorithme à 3-phases

Une analyse de ALT-GLOUTON fournit une  $(\frac{1}{2}, \frac{1}{3} + \frac{1}{r(\mathcal{M})})$ -approximation pour tout matroïde  $\mathcal{M} = (X, \mathcal{F})$  dans le cas de couleurs non-pondérées. On note qu'il est inutile de se restreindre à des matroïdes coloriés-simples. La preuve n'est pas fournie car elle suit la preuve du théorème 6.1 (page 146) et nous proposons un autre algorithme, appelé 3-PHASES, avec de meilleures garanties. En effet, nous prouvons que 3-PHASES est  $(\frac{k-1}{k}, \frac{1}{k})$ -approché pour tout entier positif  $k$  donné en entrée. Ainsi,  $k = 2$  fournit une  $(\frac{1}{2}, \frac{1}{2})$ -approximation qui améliore la garantie de ALT-GLOUTON.

L'algorithme 3-PHASES commence avec une solution vide  $F$  et l'agent poids ajoute  $\lfloor \frac{k-1}{k}L \rfloor$  éléments de façon gloutonne (en maximisant le poids). Puis l'agent couleur ajoute à  $F$  un ensemble *d'au plus*  $\lfloor \frac{L}{k} \rfloor$  nouveaux éléments de sorte que le nombre de nouvelles couleurs (c'est-à-dire les couleurs qui ne sont pas présentes dans la première phase) soit maximum. Durant la phase finale, l'agent poids complète  $F$  de façon gloutonne en ajoutant des éléments selon leur poids jusqu'à obtenir une base. Dans ce qui suit,  $B_j$  représente les éléments insérés durant la phase  $j$ . Ainsi, 3-PHASES retourne la base  $B_1 \cup B_2 \cup B_3$ .

Les premières et troisièmes phases de l'algorithme sont toutes les deux gloutonnes et polynomiales. Il n'y a que la deuxième phase de l'algorithme qui n'est pas gloutonne. Cette phase consiste à trouver un ensemble indépendant de cardinalité limitée (au plus  $\lfloor L/k \rfloor$  éléments) dans l'intersection de deux matroïdes.

Dans la deuxième phase, le premier matroïde est obtenu comme suit : considérons la contraction de  $\mathcal{M}$  par  $B_1$  notée  $\mathcal{M}/B_1$ , et la restriction de  $\mathcal{M}/B_1$  à  $X' = \{x \in X : \mathcal{L}(x) \notin \mathcal{L}(B_1)\}$  (voir page 26). Notons que  $X' \subseteq X \setminus B_1$ . Alors la structure  $\mathcal{M}_1 = (X', \mathcal{F}_1)$  où  $\mathcal{F}_1 = \{F \subseteq X' : B_1 \cup F \in \mathcal{F}\}$  définit un matroïde car la contraction et la restriction d'un matroïde est aussi un matroïde (Schrijver, 2003). Ainsi,  $\mathcal{M}_1$  est le premier matroïde. Le deuxième matroïde  $\mathcal{M}_2 = (X', \mathcal{F}_2)$  est le matroïde de partition de  $X'$  induit par les couleurs. En effet, supposons que  $\mathcal{L}(X) = \{\ell_1, \dots, \ell_p\}$ . Soit  $X_i = \{x \in X : \mathcal{L}(x) = \ell_i\}$  pour  $i = 1, \dots, p$  l'ensemble des éléments de couleur  $\ell_i$ , alors  $\mathcal{F}_2 = \{F \subseteq X' : \forall i = 1, \dots, p, |F \cap X_i| \leq 1\}$ .

La deuxième phase consiste à trouver un ensemble  $S$  de cardinalité maximum dans  $\mathcal{F}_1 \cap \mathcal{F}_2$  (ceci peut être fait en temps polynomial (Edmonds, 1979)) et si  $|S| > \lfloor L/k \rfloor$ , retenir un sous-ensemble d'uniquement  $\lfloor L/k \rfloor$  éléments de  $S$ . En utilisant l'axiome (ii) des matroïdes (voir la définition 1.12, page 24), l'ensemble

résultant est indépendant. Ainsi, la deuxième phase de 3-PHASES s'exécute en temps polynomial.

**Théorème 6.5** *Soit  $k \geq 1$  un entier positif donné en entrée, alors 3-PHASES est  $(\frac{k-1}{k}, \frac{1}{k})$ -approché pour le cas des couleurs non-pondérées.*

**Preuve.** L'algorithme est clairement  $(0, 1)$ -approché lorsque  $k = 1$  car aucun élément est choisi durant la première phase, et durant la deuxième phase on peut insérer  $\lfloor L \rfloor = L$  éléments avec des couleurs distinctes.

Maintenant, supposons que  $k \geq 2$ . Soit  $B = \{e_1, \dots, e_r\}$  une base retournée par 3-PHASES en respectant l'ordre d'insertion des éléments par l'algorithme.

Nous nous concentrons d'abord sur le rapport d'approximation  $\frac{k-1}{k}$  pour l'agent poids. Soit  $B^w = \{e_1^w, \dots, e_r^w\}$  une base de poids maximum vérifiant  $w(e_1^w) \geq w(e_2^w) \geq \dots \geq w(e_r^w)$ . En utilisant l'axiome (iii) des matroïdes, on a  $w(e_i) \geq w(e_i^w)$  pour  $i = 1, \dots, |B_1|$  et  $|B_1| = \lfloor \frac{k-1}{k} L \rfloor$ . En sommant sur  $i$ , on obtient

$$w(B_1) \geq \sum_{j=1}^{|B_1|} w(e_j^w) \quad (6.14)$$

Les éléments de  $B^w$  étant triés par poids décroissant, il s'ensuit que

$$w(e_i) \geq w(e_j^w), \quad \forall (i, j) \in \{1, \dots, |B_1|\} \times \{|B_1| + 1, \dots, |B_1| + |B_2|\} \quad (6.15)$$

$B_2$  est l'ensemble des éléments ajoutés durant la deuxième phase par l'agent couleur et  $|B_2| \leq \lfloor L/k \rfloor$ . Puisque  $|B_1| = \lfloor \frac{k-1}{k} L \rfloor \geq (k-1) \lfloor \frac{L}{k} \rfloor \geq (k-1)|B_2|$ , alors il est possible de partager  $B_1$  en  $k-1$  ensembles disjoints  $B_1^1, \dots, B_1^{k-1}$  de façon à ce que chacun contienne au moins  $|B_2|$  éléments. En utilisant (6.15), on déduit que

$$w(B_1^t) \geq \sum_{j=|B_1^t|+1}^{|B_1|+|B_2|} w(e_j^w) \quad \text{pour } t = 1, \dots, k-1$$

En additionnant ces inégalités, on obtient

$$w(B_1) = \sum_{t=1}^{k-1} w(B_1^t) \geq (k-1) \sum_{j=|B_1|+1}^{|B_1|+|B_2|} w(e_j^w)$$

D'où,

$$\frac{1}{k-1} w(B_1) \geq \sum_{j=|B_1|+1}^{|B_1|+|B_2|} w(e_j^w) \quad (6.16)$$

En additionnant (6.14) et (6.16), on obtient

$$\frac{k}{k-1} w(B_1) \geq \sum_{i=1}^{|B_1|+|B_2|} w(e_i^w) \quad (6.17)$$

Concernant la troisième phase de l'algorithme, l'axiome (iii) des matroïdes implique que

$$w(e_i) \geq w(e_i^w), \text{ pour } |B_1| + |B_2| + 1 \leq i \leq r$$

En sommant ces inégalités

$$w(B_3) = \sum_{i=|B_1|+|B_2|+1}^r w(e_i) \geq \sum_{i=|B_1|+|B_2|+1}^r w(e_i^w)$$

que l'on additionne à (6.17) pour obtenir

$$\frac{k}{k-1}w(B) \geq \frac{k}{k-1}w(B_1) + w(B_3) \geq \sum_{i=1}^{|B_1|+|B_2|} w(e_i^w) + \sum_{i=|B_1|+|B_2|+1}^r w(e_i^w) = w(B^w)$$

D'où, 3-PHASES est  $(\frac{k-1}{k}, \cdot)$ -approché.

Considérons maintenant l'agent couleur et soit  $B^{\mathcal{L}}$  une base de  $\mathcal{M}$  avec  $L$  couleurs (nombre maximum). Soit  $E^{\mathcal{L}}$  un sous-ensemble de  $B^{\mathcal{L}}$  contenant un élément par couleur dans  $B^{\mathcal{L}}$ , c'est-à-dire  $|E^{\mathcal{L}}| = |\mathcal{L}(E^{\mathcal{L}})| = |\mathcal{L}(B^{\mathcal{L}})| = L$ . La deuxième phase de l'algorithme consiste à ajouter à  $B_1$ , au plus  $\lfloor L/k \rfloor$  éléments avec des couleurs qui n'apparaissent pas dans  $\mathcal{L}(B_1)$ . Soit  $E = \{e \in E^{\mathcal{L}} : \mathcal{L}(e) \notin \mathcal{L}(B_1)\}$ . Notons  $t$  le nombre de couleurs que  $B_1$  et  $E^{\mathcal{L}}$  partagent. On a  $|\mathcal{L}(B_1)| \geq t$  et  $|E| = L - t$ .

Supposons que  $|E| \leq |B_1|$ . On obtient  $L - t = |E| \leq |B_1| = \lfloor \frac{k-1}{k}L \rfloor$ . Alors  $t \geq L - \lfloor \frac{k-1}{k}L \rfloor \geq L - \frac{k-1}{k}L = \frac{L}{k}$ . En utilisant  $|\mathcal{L}(B_1)| \geq t$ , on observe que  $B_1$ , et a fortiori  $B$ , contiennent au moins  $L/k$  couleurs.

Supposons maintenant que  $|E| > |B_1|$ . Au moins  $|E| - |B_1| = (L-t) - \lfloor \frac{k-1}{k}L \rfloor = \lceil L/k \rceil - t$  éléments avec de nouvelles couleurs sont ajoutés durant la deuxième phase. En effet, d'une part en utilisant l'axiome (iii') (définition 1.12, page 24), il existe  $A \subseteq E \setminus B_1$  avec  $|A| = |E| - |B_1|$  tel que  $A \cup B_1 \in \mathcal{F}$  (c'est-à-dire,  $A$  est un ensemble indépendant de  $\mathcal{M}_1 = (X', \mathcal{F}_1)$  où  $X' = \{x \in X : \mathcal{L}(x) \notin \mathcal{L}(B_1)\}$  et  $\mathcal{F}_1 = \{F \subseteq X' : B_1 \cup F \in \mathcal{F}\}$ ). D'autre part,  $A$  est un ensemble indépendant du matroïde de partition  $\mathcal{M}_2 = (X', \mathcal{F}_2)$  (où  $X'$  est partitionné en  $\{X_1, \dots, X_p\}$  tel que  $X_i = \{x \in X : \mathcal{L}(x) = \ell_i\}$  pour  $i = 1, \dots, p$  et  $\mathcal{F}_2 = \{F \subseteq X' : \forall i = 1, \dots, p, |F \cap X_i| \leq 1\}$ ) car  $|\mathcal{L}(A)| = |A|$ .

Ainsi,  $A$  est une solution réalisable de l'intersection des matroïdes  $\mathcal{M}_1$  et  $\mathcal{M}_2$ . Donc  $|\mathcal{L}(B_2)| \geq |\mathcal{L}(A)|$ . Puisque  $|\mathcal{L}(B_1)| \geq t$ , alors au moins  $|\mathcal{L}(B_1 \cup B_2)| = |\mathcal{L}(B_1)| + |\mathcal{L}(B_2)| \geq |\mathcal{L}(B_1)| + |\mathcal{L}(A)| = \lceil L/k \rceil \geq L/k$  couleurs apparaissent à la fin de la deuxième phase. Les éléments ajoutés durant la dernière phase ne peuvent qu'augmenter le nombre de couleurs. Dans tous les cas, l'algorithme 3-PHASES est  $(\cdot, 1/k)$ -approché.



En conclusion, 3-PHASES est  $(\frac{k-1}{k}, \frac{1}{k})$ -approché. ■

Le prochain lemme montre que les garanties de 3-PHASES sont Pareto-optimales.

**Lemme 6.1** *Soient  $\alpha, \beta \in ]0, 1[$ . Il existe des instances sans  $(\alpha, \beta)$ -approximation pour le cas des couleurs non-pondérées dans les circonstances suivantes :*

- (a)  $\alpha + \beta \geq 1$  et  $\alpha \neq \frac{k-1}{k}$  pour tout entier strictement positif  $k$  ;
- (b)  $\alpha > 0$  et  $\beta > 1/2$ .

**Preuve.** Nous prouvons le lemme sur des instances du matroïde graphique mais le résultat est valable pour des matroïdes généraux. Soient  $\alpha > 0$  et  $\beta > 0$ .

(a) Supposons que  $\alpha + \beta \geq 1$  et  $\alpha \neq \frac{k-1}{k}$  pour tout entier positif  $k$ . Il existe un unique entier  $L \geq 2$  tel que  $\frac{L-2}{L-1} < \alpha < \frac{L-1}{L}$ . Donc  $\alpha L > \frac{L(L-2)}{L-1}$  et  $(1 - \alpha)L > 1$ . Considérons le multigraphe  $G_L$ , instance du matroïde graphique avec des couleurs non-pondérées, donné dans la figure 6.4.

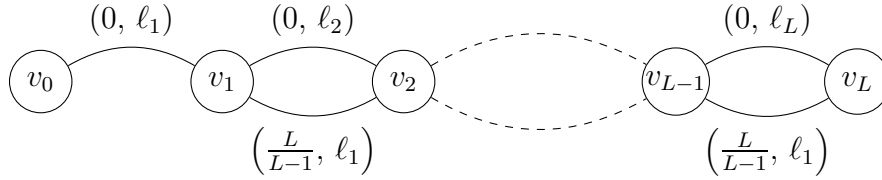


FIGURE 6.4 – Le multigraphe  $G_L$ .

Le point idéal de  $G_L$  est  $x^L = (L, L)$  en considérant les arbres définis par l'arête  $(v_0, v_1)$  et les arêtes du bas pour le poids, et les arêtes du haut pour les couleurs.

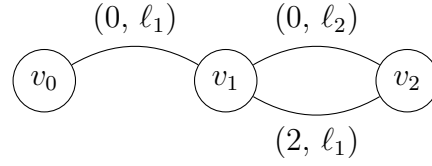
Par contradiction, supposons que  $T$  est un arbre couvrant de  $G_L$  avec un poids  $w(T) \geq \alpha L > \frac{L(L-2)}{L-1}$  et un nombre de couleurs  $|\mathcal{L}(T)| \geq (1 - \alpha)L > 1$ . Si  $T$  contient au moins  $L - 1$  arêtes de poids  $L/(L - 1)$ , alors  $T$  contient une couleur  $\ell_1$ . Ainsi,  $1 = |\mathcal{L}(T)| \geq (1 - \alpha)L > 1$ , contradiction. D'où,  $T$  contient au plus  $L - 2$  arêtes de poids  $L/(L - 1)$  et donc  $L(L - 2)/(L - 1) \geq w(T) \geq \alpha L > L(L - 2)/(L - 1)$ , contradiction.

(b) Supposons que  $\alpha > 0$  et  $\beta > 1/2$ . Considérons le multigraphe  $G_2$  représenté dans la figure 6.5.

Le multigraphe  $G_2$  admet deux différents arbres couvrants  $T_1 = \{(v_0, v_1), (v_1, v_2) \text{ haut}\}$  et  $T_2 = \{(v_0, v_1), (v_1, v_2) \text{ bas}\}$  avec des utilités respectives de  $(0, 2)$  et  $(2, 1)$ . Donc le point idéal est  $(2, 2)$ .

Il n'existe pas de solution réalisable avec un poids strictement positif et de sorte que le nombre de couleurs soit supérieur à  $2\beta > 2 * 1/2 = 1$ . ■

Le point (b) du Lemme 6.1 montre que  $\beta = 1/2$  est ce que l'on peut espérer de mieux dès lors qu'il y a une approximation non triviale pour le poids.

FIGURE 6.5 – Le multigraphe  $G_2$ .

### 6.5.2 Base de poids maximum avec un nombre exact de couleurs

Dans cette section, nous nous intéressons au cas des couleurs non-pondérées. Chaque élément du matroïde  $\mathcal{M} = (X, \mathcal{F})$  a une couleur avec un gain  $g(\ell) = 1$  pour tout  $\ell$ . Nous considérons le cas où le nombre de couleurs n'est pas fixé et nous voulons trouver une base qui maximise le poids  $w$  et qui contient exactement  $k$  couleurs où  $k \geq 1$  est un entier fixé.

Un problème connexe a été étudié par Gabow et Tarjan (1984). Ce problème consiste à trouver une base de poids minimum qui contient exactement  $q$  éléments rouges, c'est-à-dire résoudre  $\min\{w(B) : B \in \mathcal{B} \text{ et } B \text{ a } q \text{ éléments rouges}\}$  lorsqu'il y a uniquement deux couleurs, rouge ou vert. Les auteurs présentent un algorithme général efficace pour trouver une telle base. L'algorithme est polynomial sous l'hypothèse que la complexité de l'oracle d'indépendance est polynomial. Ils proposent également d'améliorer la complexité en temps pour différents matroïdes spécifiques (matroïdes graphique, transversal, couplage et partition).

Dans cette section, nous supposons qu'il y a une unique base de poids maximum, c'est-à-dire  $|\mathcal{B}_w| = 1$ . Notons que l'on peut décider quand  $|\mathcal{B}_w|$  est égal à 1 et résoudre  $\max_{B \in \mathcal{B}_w} |\mathcal{L}(B)|$  en temps polynomial (voir le théorème 6.4, page 156). L'hypothèse  $|\mathcal{B}_w| = 1$  peut toujours être satisfaite en perturbant à  $\epsilon > 0$  près les poids des éléments. Dans ce cas, le problème est  $(1 + \epsilon)$ -approché.

En utilisant les notations précédentes,  $\mathcal{B}_w = \{F_{w,g}^*\}$  puisque  $F_{w,g}^*$  est une base de  $\mathcal{B}_w$  avec un nombre maximum de couleurs distinctes. Nous nous intéressons à trouver une base de poids maximum parmi les bases qui utilisent exactement  $k$  couleurs :  $\max\{w(B) : B \in \mathcal{B} \text{ et } |\mathcal{L}(B)| = k\}$ .

Considérons les notations suivantes : soit  $\mathcal{F}_{w,k}^*$  l'ensemble des bases  $B$  qui sont maximums pour  $w$  sous la contrainte  $|\mathcal{L}(B)| = k$  et  $F_{g,w}^*$  est une base de  $\mathcal{B}_g$  de poids maximum (voir la section 6.4, page 155). Soit  $k_1^* = |\mathcal{L}(F_{w,g}^*)|$ ,  $k_2^* = |\mathcal{L}(F_{g,w}^*)|$  et  $J = \{k_1^*, \dots, k_2^*\}$ . Évidemment,  $k_1^* \leq k_2^*$  et chaque solution Pareto-optimale est dans  $\bigcup_{k \in J} \mathcal{F}_{w,k}^*$ . Soit  $N$  l'ensemble des éléments ayant une couleur qui n'est pas présente dans  $\mathcal{L}(F_{w,g}^*)$  et soit  $C$  l'ensemble des éléments restants, c'est-à-dire  $N = \{x \in B_k : \mathcal{L}(x) \notin \mathcal{L}(F_{w,g}^*)\}$  et  $C = \{x \in B_k : \mathcal{L}(x) \in \mathcal{L}(F_{w,g}^*)\}$ . Nous allons

prouver que si on décompose une base  $B_k \in \mathcal{F}_{w,k}^*$  en  $C \cup N$ , alors toutes les couleurs de  $F_{w,g}^*$  sont présentes dans  $B_k$ , chaque nouvelle couleur apparaît exactement une fois et  $B_k$  contient *exactement*  $k$  couleurs.

**Théorème 6.6** *Pour tout  $k \in J$  et pour tout  $B_k = N \cup C \in \mathcal{F}_{w,k}^*$  où  $N = \{x \in B_k : \mathcal{L}(x) \notin \mathcal{L}(F_{w,g}^*)\}$  et  $C = \{x \in B_k : \mathcal{L}(x) \in \mathcal{L}(F_{w,g}^*)\}$ , les propriétés suivantes sont valides :*

1.  $\mathcal{L}(C) = \mathcal{L}(F_{w,g}^*)$ .
2.  $|N| = |\mathcal{L}(N)|$ .
3.  $|\mathcal{L}(B_k)| = k$ .

**Preuve.** Soit  $k \in J$  et  $B_k = N \cup C \in \mathcal{F}_{w,k}^*$ .

1. Par contradiction, supposons que  $\mathcal{L}(C) \neq \mathcal{L}(F_{w,g}^*)$ . Alors par hypothèse,  $F_{w,g}^* \neq B_k$  et même  $\mathcal{L}(F_{w,g}^*) \setminus \mathcal{L}(B_k) \neq \emptyset$ . Soit  $x \in F_{w,g}^*$  avec  $\mathcal{L}(x) \notin \mathcal{L}(B_k)$ . En particulier,  $x \notin B_k$ . En utilisant la propriété d'échange d'un élément (voir le théorème 1.3 page 27), nous savons qu'il existe  $y \in B_k \setminus F_{w,g}^*$  tel que  $F' = F_{w,g}^* - x + y$  et  $B' = B_k - y + x$  sont dans  $\mathcal{B}$ . Puisque  $F_{w,g}^* \in \mathcal{B}_w$ , alors  $w(x) \geq w(y)$  et donc  $w(B') \geq w(B_k)$ . Par ailleurs, puisque  $\mathcal{L}(x) \notin \mathcal{L}(B_k)$ , alors  $|\mathcal{L}(B')| \geq |\mathcal{L}(B_k)| \geq k$ . Ainsi,  $B' \in \mathcal{F}_{w,k}^*$ , donc  $w(x) = w(y)$  car  $B_k \in \mathcal{F}_{w,k}^*$  (d'où,  $w(B_k) = w(B')$ ). On déduit que  $w(F') = w(F_{w,g}^*)$  où  $F' \neq F_{w,g}^*$ . Contradiction avec l'hypothèse initiale que  $|\mathcal{B}_w| = 1$ . D'où,  $\mathcal{L}(C) = \mathcal{L}(F_{w,g}^*)$ .

2. La preuve est similaire. Par contradiction, supposons que l'égalité n'est pas vérifiée. Donc une couleur  $\ell \in \mathcal{L}(N)$  apparaît au moins deux fois dans  $B_k$ . Soit  $y \in N$  avec  $\mathcal{L}(y) = \ell$ . Puisque  $\mathcal{L}(y) \notin \mathcal{L}(F_{w,g}^*)$ , alors  $y \notin F_{w,g}^*$ . Ainsi, en utilisant le théorème 1.3 (page 27), nous savons qu'il existe  $x \in F_{w,g}^* \setminus B_k$  tel que  $F' = F_{w,g}^* - x + y$  et  $B' = B_k - y + x$  sont dans  $\mathcal{B}$ . D'où,  $w(x) \geq w(y)$  car  $F_{w,g}^* \in \mathcal{B}_w$ . Par ailleurs,  $|\mathcal{L}(B')| \geq |\mathcal{L}(B_k)| \geq k$  car la couleur  $\ell$  apparaît deux fois dans  $B_k$ . Ainsi,  $B' \in \mathcal{F}_{w,k}^*$  et alors  $w(x) = w(y)$ . Une fois de plus, on obtient une contradiction avec l'hypothèse initiale  $|\mathcal{B}_w| = 1$ .

3. Supposons que  $|\mathcal{L}(B_k)| > k$ . En particulier, on déduit que  $k > k_1^*$  car initialement,  $|\mathcal{B}_w| = 1$ ; d'où,  $B_k \neq F_{w,g}^*$ . En utilisant les deux propriétés précédentes, on sait que  $B_k = N \cup C$  avec  $\mathcal{L}(C) = \mathcal{L}(F_{w,g}^*)$  et  $|N| = |\mathcal{L}(N)|$ . Comme précédemment, soit  $y \in N$  avec  $\mathcal{L}(y) = \ell$  et  $y \notin F_{w,g}^*$ . Il existe  $x \in F_{w,g}^* \setminus B_k$  tel que  $F' = F_{w,g}^* - x + y$  et  $B' = B_k - y + x$  sont dans  $\mathcal{B}$ . Ainsi,  $w(x) \geq w(y)$  et  $|\mathcal{L}(B')| \geq |\mathcal{L}(B_k)| - 1 \geq k$  car  $|\mathcal{L}(B_k)| > k$ . En conclusion,  $B' \in \mathcal{F}_{w,k}^*$ ,  $w(x) = w(y)$  et on obtient la même contradiction que précédemment. ■

**Remarque 6.1** *Lorsque  $|\mathcal{B}_w| > 1$ , le théorème 6.6 peut être révisé comme suit : pour tout  $k \in J$ , il existe  $B_k = N \cup C \in \mathcal{F}_{w,k}^*$  et  $B \in \mathcal{B}_w$  tels que  $\mathcal{L}(C) = \mathcal{L}(B)$ ,*

$|N| = |\mathcal{L}(N)|$  et  $|\mathcal{L}(B)| = k$  en considérant une base  $B_k \in \mathcal{F}_{w,k}^*$  qui a un nombre maximum d'éléments en commun avec une base  $B$  de  $\mathcal{B}_w$ .

En utilisant le théorème 6.6, on peut penser que l'algorithme suivant trouve une base  $B_k = N \cup C$  de  $\mathcal{F}_{w,k}^*$  pour un entier  $k \geq k_1^*$  : au début considérer l'unique base  $F_{w,g}^*$ , supprimer un ensemble  $A$  de  $k' = |N| = k - k_1^*$  éléments de  $F_{w,g}^*$  de sorte que  $\mathcal{L}(F_{w,g}^*) = \mathcal{L}(F_{w,g}^* \setminus A)$  et ajouter  $k'$  nouvelles couleurs. Malheureusement, cette procédure n'est pas valable car il existe des instances où  $C$  n'est pas forcément un sous-ensemble de  $F_{w,g}^*$  comme montré dans l'exemple 6.7.

**Exemple 6.7** Considérons le matroïde graphique colorié où  $G = (V, E)$  est un graphe connexe de 5 sommets décrits dans la figure 6.6.

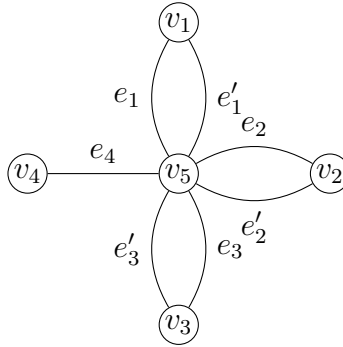


FIGURE 6.6 – Exemple 6.7.

$E = \{e_1, \dots, e_4\} \cup \{e'_1, e'_2, e'_3\}$  avec  $w(e_i) = 2$  pour  $i = 1, \dots, 4$  et  $w(e'_i) = 1$  pour  $i = 1, \dots, 3$ . Finalement,  $\mathcal{L}(e_1) = \mathcal{L}(e_4) = \ell_1$ ,  $\mathcal{L}(e_2) = \mathcal{L}(e'_1) = \ell_2$ ,  $\mathcal{L}(e_3) = \mathcal{L}(e'_2) = \ell_3$  et  $\mathcal{L}(e'_3) = \ell_4$ ; donc, nous avons 4 couleurs  $\{\ell_1, \dots, \ell_4\}$ . On obtient  $\mathcal{F}_{w,3}^* = \{F_{w,g}^*\} = \{\{e_1, \dots, e_4\}\}$  qui utilise 3 couleurs  $\{\ell_1, \ell_2, \ell_3\}$ , et a un poids total de 8, tandis que  $\mathcal{F}_{w,4}^* = \{\{e'_1, e'_2, e'_3, e_4\}\}$  utilise les 4 couleurs et a un poids total de 5.

Nous concluons cette section en proposant l'algorithme EXACT (voir l'algorithme 6.2) qui détermine une base de  $\mathcal{F}_{w,k}^*$  lorsque  $|\mathcal{B}_w| = 1$ .

En utilisant le théorème 6.6, on déduit que l'algorithme EXACT retourne une base de poids maximum parmi les bases contenant exactement  $k$  couleurs. Notons que cet algorithme ne s'exécute pas en temps polynomial.

---

**Algorithme 6.2 : EXACT**

---

**Données :** un matroïde  $\mathcal{M} = (X, \mathcal{F})$ , une fonction poids  $w : X \rightarrow \mathbb{R}^+$ , une fonction couleur  $\mathcal{L} : X \rightarrow \{\ell_1, \dots, \ell_p\}$  et un entier  $k$

**Résultat :** une base de  $\mathcal{F}_{w,k}^*$

- 1 décider si  $|\mathcal{B}_w| = 1$  et trouver  $F_{w,g}^*$
- 2  $\mathcal{C} \leftarrow \mathcal{L}(F_{w,g}^*)$
- 3 **si**  $k < k_1^* = |\mathcal{C}|$  **alors**
- 4      $\lfloor$  retourner  $F_{w,g}^*$
- 5  $B_k \leftarrow \emptyset$
- 6 **pour tout** sous-ensemble  $C \cup N \in \mathcal{F}$  avec  $N \subseteq X \setminus F_{w,g}^*$ ,  $|N| = k - k_1^*$  et  $|C| = k_1^*$  **faire**
- 7     **si**  $|N| = |\mathcal{L}(N)|$  et  $\mathcal{L}(C) = \mathcal{C}$  **alors**
- 8         soit  $\mathcal{M}' = (X', \mathcal{F}')$  le matroïde où  $X' = \{x \in X \setminus (C \cup N) : \mathcal{L}(x) \in \mathcal{C}\}$  et  $\mathcal{F}' \in \mathcal{F}$   
        **si et seulement si**  $F' \cup (C \cup N) \in \mathcal{F}$
- 9          $B'_{C \cup N} \leftarrow \text{GLOUTON}(\mathcal{M}', w)$  ( $B'_{C \cup N}$  est une base de poids maximum de  $\mathcal{M}'$ )
- 10         **si**  $w(B'_{C \cup N}) + w(C \cup N) > w(B_k)$  **alors**
- 11              $\lfloor$   $B_k \leftarrow B'_{C \cup N} \cup C \cup N$
- 12 **retourner**  $B_k$

---

## Conclusion et perspectives

Pour le problème général, nous avons proposé un algorithme polynomial  $(\frac{1}{2}, \frac{1}{4})$ -approché du point idéal pour des matroïdes coloriés-simples. Une  $(\frac{1}{3}, \frac{1}{3})$ -approximation existe pour des matroïdes simples en inversant l'ordre de passage des agents dans l'algorithme. Une question importante consiste à améliorer ces approximations. Nous conjecturons l'existence d'une  $(1/2, 1/3)$ -approximation et qu'elle est Pareto-optimale, c'est-à-dire qu'il existe des instances sans  $(\alpha, \beta)$ -approximation telle que  $\alpha \geq 1/2$  et  $\beta \geq 1/3$  avec au moins une inégalité stricte.

Pour le cas des couleurs non pondérées, nous avons proposé des algorithmes déterministes polynomiaux qui construisent plusieurs compromis et nous avons prouvé que certains compromis ne peuvent être obtenus. Notons que le résultat de complexité de la section 6.4 n'est pas valide pour le cas des couleurs non pondérées, donc sa complexité exacte est un problème ouvert. Une première étape dans cette direction serait de considérer le problème ouvert de la construction de  $F_{g,w}^*$  (une solution Pareto-optimale de poids maximum parmi les solutions de gain maximum).

Une autre perspective est de considérer des classes plus générales de fonctions objectifs. Supposons que les deux fonctions d'utilité soient de type « gain des couleurs » de sorte que les couleurs de la première fonction d'utilité soient indépendantes de celles de la deuxième. Alors une révision appropriée de l'algorithme ALT-GLOUTON fournit une  $(\frac{1}{3}, \frac{1}{4})$ -approximation pour des matroïdes coloriés-simples.

Cependant, nous pensons que cette approximation n'est pas Pareto-optimale et nous conjecturons l'existence d'une  $(\frac{1}{3}, \frac{1}{3})$ -approximation.

Si la première fonction d'utilité est « gain des couleurs » et la seconde est « nombre des couleurs », nous pouvons déduire une  $(\frac{1}{3}, \frac{1}{3} + \frac{1}{r(\mathcal{M})})$ -approximation avec la version révisée de ALT-GLOUTON. Une question intéressante est la possibilité de modifier les phases 1 et 3 de 3-PHASES dans le but de fournir de meilleures garanties.

Cependant, si la première fonction d'utilité est « nombre des couleurs » et la seconde est « gain des couleurs », nous ne pensons pas pouvoir améliorer la  $(\frac{1}{3}, \frac{1}{4})$ -approximation avec des algorithmes de choix séquentiels (*picking sequences*).

Dans tous les cas, il est préférable que l'agent couleur choisisse ses éléments après l'agent poids dans les procédures de choix séquentiel car cela permet à l'agent couleur de se garantir une couleur supplémentaire. En effet, si l'agent poids choisit un élément en premier, alors on peut compter une première couleur. Puis, l'agent couleur tentera toujours d'ajouter des éléments de nouvelles couleurs par rapport à ce qu'a pris l'agent poids.

Finalement, il serait intéressant d'étudier le problème avec des fonctions d'utilité sous-modulaires générales.



# Conclusion et perspectives

## Conclusion

Nous nous sommes intéressés dans cette thèse à la problématique de la décision collective. Il s'agit de déterminer une solution de compromis parmi plusieurs alternatives, qui sont soumises à de multiples points de vue. Les problèmes considérés sont de nature combinatoire, ils se modélisent à l'aide d'un matroïde où une solution réalisable est une base du matroïde. Les points de vues s'expriment sous forme de fonctions d'utilité à maximiser. La solution déterminée doit être un compromis pour les membres de la collectivité et doit satisfaire, au mieux, les attentes de chacun.

L'objectif est de trouver une base d'un matroïde évaluée par plusieurs agents au moyen de fonctions d'utilité additives. Cette base représente la solution de compromis que l'on partage sur les agents de façon équitable.

Une caractéristique importante d'une solution de compromis est son équité par rapport à tous les membres de la collectivité. À cette fin, nous nous sommes inspirés des concepts d'équité utilisés dans le problème classique d'allocation de ressources pour déterminer un partage équitable de la solution de compromis sur les agents. Une première étape était d'adapter les notions d'équité classiques au problème de la base d'un matroïde multi-agents. Cela a constitué les travaux du chapitre 3 pour l'équitabilité, l'absence de jalousie (que nous avons introduit), la proportionnalité et l'absence d'envie. Il est bien connu, pour ces notions d'équité, qu'il n'existe pas toujours d'allocation de biens indivisibles qui les satisfasse, ou encore déterminer l'existence d'une telle allocation est NP-complet. Ces constats négatifs sont de fait valables pour le problème de la base d'un matroïde multi-agents. Nous avons proposé de relaxer ces notions d'équité afin de garantir l'existence d'une base qui satisfait le critère d'équité relaxé. Nous avons proposé le concept de solution *proche de l'équité*, qui signifie satisfaire l'équité à un élément près. Cela a permis de garantir l'existence d'une solution proche de l'équité et de sa construction en temps polynomial pour l'absence de jalousie, la proportionnalité et l'absence d'envie (pour deux agents).

Par la suite, nous nous sommes intéressés à fournir une solution de compromis avec une garantie, dans le pire cas, sur l'utilité de chaque agent. D'un point de vue



multi-objectif, ce problème consiste à trouver une solution réalisable qui approche le point idéal du problème avec une garantie de performance.

Nous nous sommes d'abord intéressés au cas général de  $n \geq 2$  agents avec des utilités additives (problème de la base d'un matroïde multi-agents). Il s'avère que ce problème n'admet pas d'approximation à rapports constants, même pour deux agents. Comme alternative, nous avons proposé des approximations dépendant du paramètre  $\alpha_i$ , représentant l'utilité maximum de l'agent  $i$  pour un élément du matroïde. Nous avons proposé un algorithme déterministe polynomial centralisé pour tout  $n \geq 1$  et un autre décentralisé pour  $n \leq 9$  agents qui construisent une base d'un matroïde avec une borne inférieure sur l'utilité de chaque agent. L'idée sous-jacente est qu'une solution issue d'un protocole décentralisé est plus facilement acceptée par l'ensemble des agents. Ceux-ci prenant part plus activement à la création de la solution finale. L'algorithme centralisé est  $(W_n(\alpha_i))_{i=1,\dots,n}$ -approché et l'algorithme décentralisé est  $(\frac{i}{n}V_i(\frac{n}{i}\alpha_i))_{i=1,\dots,n}$ -approché. Nous nous sommes intéressés à l'application de ce problème au problème classique d'allocation de biens indivisibles. Les garanties de nos algorithmes améliorent la garantie connue pour le problème d'allocation de biens indivisibles qui est une  $(V_n(\alpha_i))_{i=1,\dots,n}$ -approximation. En effet,  $W_n(\alpha_i) \geq V_n(\alpha_i)$  pour tout  $\alpha_i \in [0, 1]$  (voir la propriété 4.1, page 103). De plus, nous savons grâce au Lemme 2.2 (page 66) que  $\frac{i}{n}V_i(\frac{n}{i}\alpha_i) \geq V_n(\alpha_i)$  pour  $i = 1, \dots, n - 1$  et  $\alpha_i \in [0, 1]$ . Nous avons également proposé une borne supérieure sur l'utilité de l'agent le plus défavorisé. Ces bornes inférieure et supérieure coïncident pour certaines valeurs de  $\alpha_i \in [0, 1]$ .

Notons, contre toute attente, que la fonction  $W_n$  est non-monotone. La borne supérieure  $\overline{W}_n(\alpha)$  sur l'utilité de l'agent le plus défavorisé (voir la section 4.4, page 114) vient confirmer la non-monotonie de cette garantie. L'idée est que de façon générale, plus  $\alpha_i$  est petit, meilleur est le partage. Ceci est capturé par la fonction  $V_n$ . Cependant, lorsque l'on regarde le problème plus en détails, la fonction n'est pas monotone, ce qui est le cas de  $W_n$ . Ceci peut s'expliquer par l'aspect indivisible des éléments où certaines valeurs de  $\alpha_i$  permettent d'atteindre de meilleures garanties. Par exemple, pour  $n = 2$  agents, si  $\alpha_1 = \alpha_2 = 0.5$  alors chaque agent peut avoir  $W_2(\alpha_1) = W_2(\alpha_2) = 0.5$  alors que si  $\alpha_1 = \alpha_2 = 0.4$ , il existe des instances où les agents ne peuvent avoir tous les deux plus de  $W_2(\alpha_1) = W_2(\alpha_2) = 0.4$  (voir les instances proposées dans la section 4.4, page 114). On peut noter que bon nombre de problèmes discrets sont le sujet d'*anomalies* comme en ordonnancement (Graham, 1969) et en transport (paradoxes de Pigou (1924); Braess (1968); Fisk (1979)).

Dans l'optique de généraliser l'additivité des fonctions d'utilité, nous nous sommes intéressés à un autre problème défini sur un matroïde qui est évalué par deux agents avec différentes fonctions d'utilité. La première fonction d'utilité est additive et la seconde est une fonction sous-modulaire particulière re-

présentée par des couleurs. Cette fonction est appelée coloration pondérée (*the chromatic price*). Nous montrons qu'il est possible d'obtenir une solution Pareto-optimale lexicographique du problème en temps polynomial (théorème 6.4, page 156). Il s'agit de la base de coût des couleurs maximum parmi celles de poids maximum. Si on s'intéresse à la classe générale des matroïdes, ce problème n'admet pas d'approximation à rapports constants. Cependant, si on considère des classes particulières de matroïdes, alors nous pouvons obtenir des garanties intéressantes en utilisant des algorithmes de choix séquentiel. Nous avons proposé un premier algorithme polynomial  $(\frac{1}{2}, \frac{1}{4})$ -approché du point idéal pour des matroïdes colorie-simples. En inversant l'ordre des agents dans l'algorithme, nous obtenons une  $(\frac{1}{3}, \frac{1}{3})$ -approximation pour des matroïdes simples qui est incomparable à la garantie initiale. Lorsque chaque couleur apparaît au plus une fois sur les éléments du matroïde, la fonction coloration pondérée devient additive. Dans ce cas et pour des matroïdes  $(x + 1)$ -simples, le problème est  $(\frac{x}{x+y}, \frac{y}{x+y+xy})$ -approché du point idéal pour tout  $x, y \geq 1$ . Lorsque toutes les couleurs ont le même coût, nous proposons un algorithme  $(\frac{k-1}{k}, \frac{1}{k})$ -approché pour tout  $k \geq 1$  donné en entrée. Nous avons également pu observer que certains rapports d'approximation ne peuvent être obtenus du fait de la non-existence de solutions dans certaines instances.

## Perspectives

L'objectif de cette thèse était, d'une part de généraliser, et d'autre part d'améliorer certains résultats obtenus pour le problème classique d'allocation de biens indivisibles, à une structure plus générale définie par un matroïde. Afin d'élargir le champ d'application des approches et des techniques utilisées, de nombreuses perspectives de recherche se dégagent de ce travail.

### 1. Amélioration des résultats obtenus

Les résultats d'approximation présentés dans cette thèse ne sont pas tous prouvés comme étant les meilleurs possibles. Par exemple, la garantie  $(W_n(\alpha_i))_{i \in N}$  pour le problème de la base multi-agents avec des fonctions d'utilité additives, ou encore le problème bi-objectif avec des utilités hétérogènes (étudié dans le chapitre 6) qui est  $(\frac{1}{2}, \frac{1}{4})$ -approché (nous conjecturons qu'une  $(\frac{1}{2}, \frac{1}{3})$ -approximation existe). Une perspective serait, soit d'améliorer ces garanties, soit d'exhiber des instances pour lesquelles il n'est pas possible d'améliorer la garantie proposée.

Par ailleurs, il serait intéressant de généraliser pour tout  $n \geq 2$  les protocoles décentralisés proposés dans les chapitres 2 (pour trouver une solution proche de l'absence d'envie) et 5 (pour trouver une garantie dans le pire cas sur l'utilité de chaque agent).

## 2. Généralisation de la structure du problème

Rappelons que le problème générique d'allocation de ressources est défini sur un ensemble d'éléments  $R$  et une collection  $\mathcal{R} \subseteq 2^R$  d'allocations admissibles. Nous avons traité dans cette thèse le cas où  $\mathcal{R}$  contient tous les ensembles indépendants d'un matroïde. De plus, dans la majorité des cas, nous avons supposé que les utilités des agents sont additives. Une autre perspective serait d'aller plus loin dans la généralisation de ce problème selon les pistes suivantes :

### Généraliser la collection des allocations admissibles

Bien que les matroïdes soient connus pour modéliser plusieurs problèmes de nature combinatoire, on peut pousser la généralisation de la collection  $\mathcal{R}$  au delà des matroïdes et l'enrichir par l'une des structures suivantes :

- Problème d'intersection de matroïdes : définissons  $n$  matroïdes pondérés  $\mathcal{M}_1 = (R, \mathcal{F}_1, u_1), \dots, \mathcal{M}_n = (R, \mathcal{F}_n, u_n)$  et considérons le problème qui consiste à trouver un ensemble indépendant  $F \in \mathcal{F}_1 \cap \dots \cap \mathcal{F}_n$ , c'est-à-dire que la collection  $\mathcal{R}$  est définie par  $\mathcal{R} = \{F \subseteq R : F \in \mathcal{F}_1 \cap \dots \cap \mathcal{F}_n\}$ . Notons que le calcul d'un ensemble indépendant de poids maximum commun à  $n = 2$  matroïdes est polynomial, tandis que pour  $n \geq 3$ , il est NP-difficile.
- Les *systèmes d'indépendance* vérifient uniquement les axiomes (i) et (ii) de la définition 1.12 des matroïdes (page 24). Ce système d'ensembles permet également de généraliser les problèmes du stable de poids maximum, du plus court chemin, du voyageur de commerce, du sac à dos, de l'arbre de Steiner, etc. (Korte et Vygen, 2008).
- Les *greedoïdes* vérifient uniquement les axiomes (i) et (iii) de la définition 1.12 des matroïdes (page 24). Les greedoïdes généralisent aussi le problème de la couverture des sommets d'un graphe (Korte et Vygen, 2008).
- Les *polymatroïdes* sont des polytopes définis par  $P(f) = \{x \in (\mathbb{R}^+)^X : x \geq 0, \sum_{i \in F} x_i \leq f(F), \forall F \subseteq X\}$  où  $X$  est un ensemble fini d'éléments et  $f : 2^X \rightarrow \mathbb{R}^+$  est une fonction sous-modulaire. Un matroïde  $\mathcal{M} = (X, \mathcal{F})$  est un cas particulier où  $f$  correspond à la fonction rang  $r$  du matroïde  $\mathcal{M}$  (voir le théorème 1.2, page 26). L'intersection de deux matroïdes  $\mathcal{M}_1 = (X, \mathcal{F}_1)$  et  $\mathcal{M}_2 = (X, \mathcal{F}_2)$  peut également être généralisée par un polymatroïde en posant  $f(F) = r_1(F) + r_2(X \setminus F)$  pour tout  $F \subseteq X$  où  $r_i$  est le rang de  $\mathcal{M}_i$ ,  $i = 1, 2$ . D'autres exemples de polymatroïdes sont proposés par Schrijver (2003).

## Généraliser l'additivité des fonctions d'utilité des agents

Par exemple, considérer des fonctions d'utilité *k-additives* dont la définition suivante est inspirée des travaux de Grabisch (1997). Une fonction d'utilité  $u$  est dite *k-additive*<sup>1</sup> pour un certain  $k \in \{1, \dots, |R|\}$  si et seulement s'il existe un ensemble de coefficients  $\{\alpha_i^F : F \subseteq R\}$  tels que  $\alpha_i^F = 0$  lorsque  $|F| > k$ , et pour tout  $R' \subseteq R$ ,  $u(R') = \sum_{F \subseteq R'} \alpha_F$  (Chevaleyre *et al.*, 2008). Cela signifie que l'utilité de l'ensemble  $R'$  est la somme des utilités marginales attribuées à des sous-ensembles  $F$  de  $R'$  de taille au plus  $k$ . L'additivité est le cas particulier où  $k = 1$ . Chevaleyre *et al.* (2008); Nguyen *et al.* (2013) traitent le problème classique d'allocation de biens indivisibles avec des fonctions d'utilité *k-additives*.

Il est également possible de généraliser les fonctions d'utilité additives par des fonctions d'utilité *sous-additives* définies comme suit. Une fonction d'utilité  $u : 2^R \rightarrow \mathbb{R}$  est sous-additive si pour tout  $F_1, F_2 \subseteq R$ ,  $u(F_1) + u(F_2) \geq u(F_1 \cup F_2)$ . Ces fonctions d'utilité sont étudiées par Dobzinski *et al.* (2005); Chekuri *et al.* (2010) pour le problème classique d'allocation de biens indivisibles.

On peut également considérer des fonctions d'utilité sous-modulaires, c'est-à-dire,  $u(F_1) + u(F_2) \geq u(F_1 \cup F_2) + u(F_1 \cap F_2)$  pour tout  $F_1, F_2 \subseteq R$ . La sous-modularité permet également de généraliser les fonctions d'utilité pondérées colorées. La sous-modularité est abordée notamment dans Lehmann *et al.* (2001); Golovin (2005); Vondrák (2008); Goemans *et al.* (2009); Chekuri *et al.* (2010) pour le problème classique d'allocation de biens indivisibles. Nous avons commencé à étudier un type particulier de fonction d'utilité sous-modulaire défini par des couleurs pour le cas de deux agents (chapitre 6). L'idée serait de généraliser le problème, d'une part, avec un matroïde évalué par cette fonction d'utilité pour  $n \geq 2$  et d'autre part, par des fonctions d'utilité sous-modulaires générales.

Un récapitulatif de tous les résultats obtenus selon ces différentes fonctions d'utilités pour le problème classique d'allocation de biens indivisibles est proposé par Nguyen *et al.* (2013).

Dans certains SMA, il peut être difficile pour les agents d'éliciter des préférences numériques sur les éléments (ou biens indivisibles). Dans ce cas, il serait intéressant de considérer des utilités ordinales plutôt que cardinales sur les éléments ou des portions de la ressource. Il existe quelques travaux qui traitent du problème classique d'allocation de biens indivisibles avec des utilités ordinales

---

1. Formellement, la *k-additivité* est définie en utilisant l'inversion de Möbius. L'inversion de Möbius associé à une fonction d'ensembles  $u : 2^R \rightarrow \mathbb{R}$  est une fonction  $m : 2^R \rightarrow \mathbb{R}$  définie par  $m(R') = \sum_{F \subseteq R'} (-1)^{|R' \setminus F|} u(F)$  pour tout  $R' \subseteq R$ . Ainsi,  $u$  est *k-additive* ( $k \in \{1, \dots, |R|\}$ ) si son inversion de Möbius est nul pour tout  $E \subseteq R$  tel que  $|E| > k$ , et il existe un sous-ensemble  $F \subseteq R$  avec  $|F| = k$  et dont l'inversion de Möbius est non nul Grabisch (1997).

(Herreiner et Puppe, 2002; Brams *et al.*, 2003; Bouveret *et al.*, 2010; Aziz *et al.*, 2014; Baumeister *et al.*, 2014).

### 3. Équité de la solution de compromis

Nous avons pu constater le nombre important de variantes du problème classique d'allocation de biens indivisibles selon le sens prêté à l'équité, en définissant plusieurs notions de bien-être social comme celles exposées dans la section 2.2 (page 54). Il serait intéressant d'appliquer d'autres critères d'équité, comme le critère égalitariste, la juste part max-min, l'opérateur OWA, etc. pour partager une base d'un matroïde sur plusieurs agents.

Nous avons également constaté que pour la plupart des notions d'équité définies dans la section 2.2 pour le problème classique d'allocation de biens indivisibles, il n'existe pas toujours une allocation qui satisfasse ce critère, ou encore déterminer l'existence d'une telle allocation est un problème NP-complet. Ces constats négatifs sont de fait valables pour toute généralisation de la collection  $\mathcal{R}$  d'allocations admissibles. Une perspective serait de relâcher les définitions de l'équité afin de garantir l'existence d'allocations qui satisfont le critère d'équité relâché et de réduire la complexité de calcul de telles solutions. Une première idée avait été de considérer des solutions qui satisfont un critère d'équité à un élément près dans le chapitre 3. Il serait intéressant de proposer d'autres relaxations de l'équité.

En outre, nous avons supposé dans cette thèse que chaque agent a une vue globale sur les autres agents, c'est-à-dire que chaque agent a accès aux mêmes informations par rapport à tous les autres agents. Une situation intéressante serait de supposer que les agents ont une vision locale et non globale, c'est-à-dire que les agents ne peuvent interagir qu'avec certains (leurs voisins). L'idée consiste à créer un graphe orienté où les sommets sont les agents et il existe un arc qui va d'un sommet  $a$  vers un sommet  $b$  si l'agent  $a$  peut interagir avec l'agent  $b$ . Il s'agirait alors de résoudre les problèmes d'allocation de ressources en tenant compte de la structure du graphe des interactions possibles entre les agents. Si le graphe est complet alors tous les agents peuvent interagir les uns avec les autres, qui est l'hypothèse que nous avons fait dans cette thèse. Cette problématique a été étudiée par Chevaleyre *et al.* (2007c) qui définissent un graphe non orienté où les arêtes représentent la possibilité que deux agents négocient (échange de ressources) en cas d'envie. Ils proposent une topologie de négociation pour éliminer l'envie. Ce type de modèles peut avoir des applications dans les réseaux sociaux en s'intéressant aux interactions d'un agent avec ses voisins.

## Troisième partie

### Annexes



## Preuve du lemme 2.2

**Lemme 2.2.** (page 66) *Étant donnés deux entiers  $n, k$  tels que  $n \geq 2$  et  $n > k \geq 1$ , et un réel  $x \in I(n, p) \cup NI(n, p)$  où  $p \geq 1$  est un entier, alors*

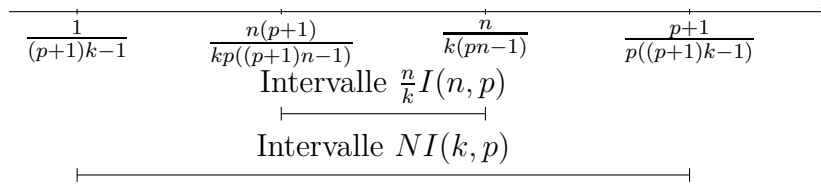
I.  $\frac{k}{n}V_k\left(\frac{n}{k}x\right) \geq V_n(x) + Z_{n,k}(x) \geq V_n(x)$  ;

II.  $\frac{k}{n}V_k\left(\frac{n}{k}x\right) \geq \frac{k+1}{n}V_{k+1}\left(\frac{n}{k+1}x\right)$ .

**Preuve.** Pour le point I., si  $x \in \left[\frac{1}{n-1}, 1\right]$ , alors  $\frac{k}{n}V_k\left(\frac{n}{k}x\right) \geq V_n(x) + Z_{n,k}(x) = 0$ . Sinon, soit  $x \in I(n, p) \cup NI(n, p)$  pour des entiers  $n \geq 2$  et  $p \geq 1$ .

1. Si  $x \in I(n, p) = \left[\frac{p+1}{p((p+1)n-1)}, \frac{1}{pn-1}\right]$  alors  $V_n(x) = 1 - p(n-1)x$ . Ainsi,  $\frac{n}{k}x \in \frac{n}{k}I(n, p) = \left[\frac{n(p+1)}{kp((p+1)n-1)}, \frac{n}{k(pn-1)}\right]$ . Le but est de situer les valeurs de  $\frac{n}{k}x$  dans les intervalles  $NI(k, p)$  et  $I(k, p)$  comme dans la définition 2.3 (page 63). On distingue deux cas :

- (a) Si  $p \geq \frac{k}{n-k}$  alors il en résulte l'axe suivant et en dessous, les extrémités des intervalles  $\frac{n}{k}I(n, p)$  et  $NI(k, p)$ .



En effet,

- $\frac{1}{(p+1)k-1} \leq \frac{n(p+1)}{kp((p+1)n-1)}$  si et seulement si  $p \geq \frac{-n(k-1)}{kn-n+k}$  qui est toujours vrai car  $n > k \geq 1$ ,  $n \geq 2$  et  $p \geq 1$ .
- $\frac{n(p+1)}{kp((p+1)n-1)} \leq \frac{n}{k(pn-1)}$  si et seulement si  $1 \geq 0$ .
- $\frac{n}{k(pn-1)} \leq \frac{p+1}{p((p+1)k-1)}$  si et seulement si  $p \geq \frac{k}{n-k}$  qui est vrai par hypothèse.

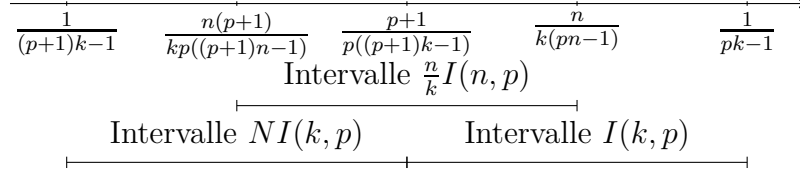
Puisque  $\frac{n}{k}x \in \frac{n}{k}I(n, p) \subset NI(k, p) = \left]\frac{1}{(p+1)k-1}, \frac{p+1}{p((p+1)k-1)}\right[$ , alors

$$\frac{k}{n}V_k\left(\frac{n}{k}x\right) = \frac{k}{n} \frac{p}{(p+1)k-1} \geq V_n(x) + Z_{n,k}(x) = 1 - p(n-1)x + \frac{(n-k)p^2x}{n(p+1)((p+1)k-1)}$$

si et seulement si  $x \geq \frac{p+1}{p((p+1)n-1)}$  qui est vrai.



- (b) Si  $p < \frac{k}{n-k}$  alors il en résulte l'axe suivant et en dessous, les extrémités des intervalles  $\frac{n}{k}I(n, p)$ ,  $NI(k, p)$  et  $I(k, p)$ .



- $\frac{1}{(p+1)^{k-1}} \leq \frac{n(p+1)}{kp((p+1)^{n-1})}$  si et seulement si  $p \geq \frac{-n(k-1)}{kn-n+k}$  qui est toujours vrai car  $n > k \geq 1$ ,  $n \geq 2$  et  $p \geq 1$ .
- $\frac{n(p+1)}{kp((p+1)^{n-1})} < \frac{p+1}{p((p+1)^{k-1})}$  si et seulement si  $n > k$  qui est vrai.
- $\frac{p+1}{p((p+1)^{k-1})} < \frac{n}{k(pn-1)}$  si et seulement si  $p < \frac{k}{n-k}$  qui est vrai par hypothèse.
- $\frac{n}{k(pn-1)} < \frac{1}{pk-1}$  si et seulement si  $n > k$ .

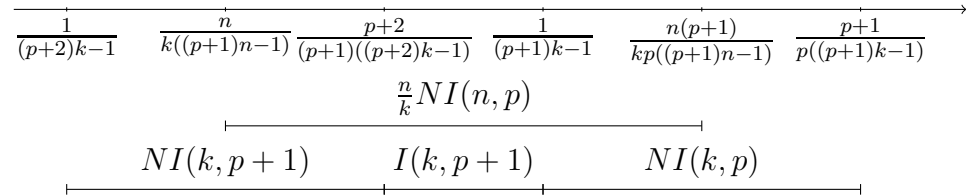
i. Si  $\frac{n}{k}x \in \left[ \frac{n(p+1)}{kp((p+1)^{n-1})}, \frac{p+1}{p((p+1)^{k-1})} \right] \subset NI(k, p) = \left[ \frac{1}{(p+1)^{k-1}}, \frac{p+1}{p((p+1)^{k-1})} \right]$  alors  $\frac{k}{n}V_k\left(\frac{n}{k}x\right) = \frac{k}{n}\frac{p}{(p+1)^{k-1}} \geq V_n(x) + Z_{n,k}(x) = 1 - p(n-1)x + \frac{(n-k)p^2x}{n(p+1)((p+1)^{k-1})}$  si et seulement si  $x \geq \frac{p+1}{p((p+1)^{n-1})}$  qui est vrai.

ii. Si  $\frac{n}{k}x \in \left[ \frac{p+1}{p((p+1)^{k-1})}, \frac{n}{k(pn-1)} \right] \subset I(k, p) = \left[ \frac{p+1}{p((p+1)^{k-1})}, \frac{1}{pk-1} \right]$  alors  $\frac{k}{n}V_k\left(\frac{n}{k}x\right) = \frac{k}{n}(1 - p(k-1)\frac{n}{k}x) \geq V_n(x) + Z_{n,k}(x) = 1 - p(n-1)x + \frac{(n-k)p^2x}{n(p+1)((p+1)^{k-1})}$  si et seulement si  $x \geq \frac{(p+1)((p+1)^{k-1})}{p(nkp^2 + (2nk-n-1)p + n(k-1))}$  qui est vrai car  $x \geq \frac{k(p+1)}{np((p+1)^{k-1})} \geq \frac{(p+1)((p+1)^{k-1})}{p(nkp^2 + (2nk-n-1)p + n(k-1))}$  si et seulement si  $p \geq \frac{-n(k-1)}{k(n-1)}$  qui est vrai car  $n \geq 2$  et  $1 \leq k < n$ .

2. Si  $x \in NI(n, p) = \left[ \frac{1}{(p+1)^{n-1}}, \frac{p+1}{p((p+1)^{n-1})} \right]$  alors  $V_n(x) = \frac{p}{(p+1)^{n-1}}$ .

Ainsi,  $\frac{n}{k}x \in \frac{n}{k}NI(n, p) = \left[ \frac{n}{k((p+1)^{n-1})}, \frac{n(p+1)}{kp((p+1)^{n-1})} \right]$ . Nous voulons situer les valeurs de  $\frac{n}{k}x$ . On distingue deux cas :

- (a) Si  $p \geq \frac{2k-n}{n-k}$  alors nous avons l'axe suivant et en dessous, les extrémités des intervalles  $\frac{n}{k}NI(n, p)$ ,  $NI(k, p+1)$ ,  $I(k, p+1)$  et  $NI(k, p)$ .



- $\frac{1}{(p+2)^{k-1}} \leq \frac{n}{k((p+1)^{n-1})}$  si et seulement si  $nk - n + k \geq 0$  qui est vrai puisque  $n > k \geq 1$ .
- $\frac{n}{k((p+1)^{n-1})} \leq \frac{p+2}{(p+1)((p+2)^{k-1})}$  si et seulement si  $p \geq \frac{2k-n}{n-k}$  qui est vrai par hypothèse.

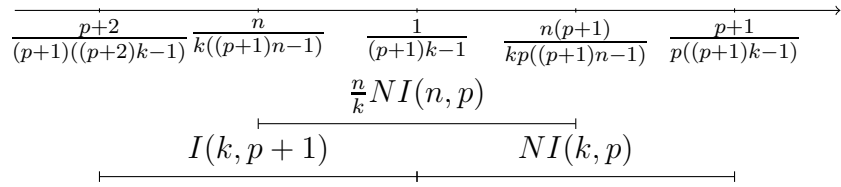
- $\frac{p+2}{(p+1)((p+2)k-1)} \leq \frac{1}{(p+1)k-1}$  si et seulement si  $1 \geq 0$ .
- $\frac{1}{(p+1)k-1} \leq \frac{n(p+1)}{kp((p+1)n-1)}$  si et seulement si  $p \geq \frac{-n(k-1)}{nk-n+k}$ , cela est vrai puisque  $n > k \geq 1$ .
- $\frac{n(p+1)}{kp((p+1)n-1)} < \frac{p+1}{p((p+1)k-1)}$  si et seulement si  $n > k$  qui est vrai par hypothèse.

i. Si  $\frac{n}{k}x \in \left] \frac{n}{k((p+1)n-1)}, \frac{p+2}{(p+1)((p+2)k-1)} \right[ \subset NI(k, p+1) = \left] \frac{1}{(p+2)k-1}, \frac{p+2}{(p+1)((p+2)k-1)} \right[$  alors  $\frac{k}{n}V_k\left(\frac{n}{k}x\right) = \frac{k}{n} \frac{p+1}{(p+2)k-1} \geq V_n(x) + Z_{n,k}(x) = \frac{p}{(p+1)n-1} + \frac{(n-k)p^2x}{n(p+1)((p+1)k-1)}$  si et seulement si  $x \leq \frac{(p+1)((p+1)k-1)((n-k)p+k(n-1))}{p^2(n-k)((p+1)n-1)((p+2)k-1)}$ . Cependant,  $x < \frac{p+1}{p((p+1)n-1)} \leq \frac{(p+1)((p+1)k-1)((n-k)p+k(n-1))}{p^2(n-k)((p+1)n-1)((p+2)k-1)}$  si et seulement si  $p \geq \frac{-(n-1)}{n}$  qui est vrai car  $n \geq 2$ .

ii. Si  $\frac{n}{k}x \in \left[ \frac{p+2}{(p+1)((p+2)k-1)}, \frac{1}{(p+1)k-1} \right] \subset I(k, p+1) = \left[ \frac{p+2}{(p+1)((p+2)k-1)}, \frac{1}{(p+1)k-1} \right]$  alors  $\frac{k}{n}V_k\left(\frac{n}{k}x\right) = \frac{k}{n}(1 - (p+1)(k-1)\frac{n}{k}x) \geq V_n(x) + \frac{x}{a}$  où  $a = \frac{n(p+1)((p+1)k-1)}{(n-k)p^2}$  i.e.  $Z_{n,k}(x) = x/a$ . Alors,  $\frac{k}{n}(1 - (p+1)(k-1)\frac{n}{k}x) \geq \frac{p}{(p+1)n-1} + \frac{x}{a}$  si et seulement si  $x \leq \frac{a(n(k-1)p+k(n-1))}{n((p+1)n-1)(a(p+1)(k-1)+1)}$ . l'inégalité  $x \leq \frac{k}{n((p+1)k-1)} \leq \frac{a(n(k-1)p+k(n-1))}{n((p+1)n-1)(a(p+1)(k-1)+1)}$  est vérifiée si et seulement si  $a \geq \frac{k((p+1)n-1)}{(n-k)p}$  et  $a = \frac{n(p+1)((p+1)k-1)}{(n-k)p^2} \geq \frac{k((p+1)n-1)}{(n-k)p}$  si et seulement si  $p \geq \frac{-n(k-1)}{nk-n+k}$  qui est toujours vrai car  $n > k \geq 1$ .

iii. Si  $\frac{n}{k}x \in \left] \frac{1}{(p+1)k-1}, \frac{n(p+1)}{kp((p+1)n-1)} \right[ \subset NI(k, p) = \left] \frac{1}{(p+1)k-1}, \frac{p+1}{p((p+1)k-1)} \right[$  alors  $\frac{k}{n}V_k\left(\frac{n}{k}x\right) = \frac{k}{n} \frac{p}{(p+1)k-1} > V_n(x) + Z_{n,k}(x) = \frac{p}{(p+1)n-1} + \frac{(n-k)p^2x}{n(p+1)((p+1)k-1)}$  si et seulement si  $x < \frac{p+1}{p((p+1)n-1)}$  qui est vrai.

(b) Si  $p < \frac{2k-n}{n-k}$  alors nous avons l'axe suivant et en dessous, les extrémités des intervalles  $\frac{n}{k}NI(n, p)$ ,  $I(k, p+1)$  et  $NI(k, p)$ .



- $\frac{p+2}{(p+1)((p+2)k-1)} < \frac{n}{k((p+1)n-1)}$  si et seulement si  $p < \frac{2k-n}{n-k}$  qui est vrai par hypothèse.
- $\frac{n}{k((p+1)n-1)} < \frac{1}{(p+1)k-1}$  si et seulement si  $n > k$  qui est toujours vrai.
- $\frac{1}{(p+1)k-1} \leq \frac{n(p+1)}{kp((p+1)n-1)}$  si et seulement si  $p \geq \frac{-n(k-1)}{nk-n+k}$  qui est vrai car  $n > k \geq 1$ .

- $\frac{n(p+1)}{kp((p+1)n-1)} < \frac{p+1}{p((p+1)k-1)}$  si et seulement si  $n > k$  qui est vrai par hypothèse.
- i. Si  $\frac{n}{k}x \in \left] \frac{n}{k((p+1)n-1)}, \frac{1}{(p+1)k-1} \right] \subset I(k, p+1)$  alors c'est le même cas que 2(a)ii.
- ii. Si  $\frac{n}{k}x \in \left] \frac{1}{(p+1)k-1}, \frac{n(p+1)}{kp((p+1)n-1)} \right[ \subset NI(k, p)$  alors c'est le même cas que 2(a)iii.

Le point II. est équivalent à montrer que  $\frac{k}{k+1}V_k\left(\frac{n}{k}x\right) \geq V_{k+1}\left(\frac{n}{k+1}x\right)$ . Soit  $y = \frac{n}{k+1}x$ , alors  $\frac{n}{k}x = \frac{k+1}{k}y$ . Ainsi, en appliquant le point I. du Lemme 2.2 pour la variable  $y$  avec  $n = k+1$ , on obtient  $\frac{k}{k+1}V_k\left(\frac{k+1}{k}y\right) \geq V_{k+1}(y)$ . En remplaçant  $y$  par sa valeur, on obtient le résultat attendu. ■

## Preuves des propriétés 4.1, 4.2 et 4.3

### B.1 Preuve de la propriété 4.1

**Propriété 4.1.** (page 103) *Étant donnés des entiers  $n \geq 2$  et  $p \geq 1$ , alors pour tout  $x \in [0, 1]$ ,*

$$V_n(x) \leq W_n(x) \leq \frac{1}{n}$$

**Preuve.** En se basant sur les différentes valeurs de  $x \in [0, 1]$  :

- Si  $x \in I_{(n,0)} = \left[ \frac{1}{n-1}, 1 \right]$  alors  $W_n(x) = V_n(x) = 0 \leq 1/n$  pour tout  $n \geq 2$ .
- Si  $x \in I_{(n,p)}^1 = \left[ \frac{p+1}{p((p+1)n-1)}, \frac{1}{np-1} \right]$  pour des entiers  $n \geq 2$  et  $p \geq 1$  alors  $W_n(x) = 1 - p(n-1)x = V_n(x)$ . De plus,  $W_n(x) = 1 - p(n-1)x \leq 1/n$  si et seulement si  $x \geq \frac{1}{np}$ . Par hypothèse,  $x \geq \frac{p+1}{p((p+1)n-1)} \geq \frac{1}{np}$  qui est vrai.
- Si  $x \in I_{(n,p)}^2 = \left[ \frac{p^2}{np^3-p^2+p+n-2}, \frac{p+1}{p((p+1)n-1)} \right]$  pour des entiers  $n \geq 2$  et  $p \geq 1$  alors  $W_n(x) = \frac{p(1-px)}{(n-1)p+n-2} \geq W_n\left(\frac{p+1}{p((p+1)n-1)}\right) = V_n(x)$  car  $W_n$  est strictement décroissante sur  $I_{(n,p)}^2$ . De plus,  $W_n(x) = \frac{p(1-px)}{(n-1)p+n-2} \leq 1/n$  si et seulement si  $x \geq \frac{p-n+2}{np^2}$ . Par hypothèse,  $x \geq \frac{p^2}{np^3-p^2+p+n-2} \geq \frac{p-n+2}{np^2}$  si et seulement si  $(n-1)^2p^3 - (n-1)p^2 + (n-2)^2 \geq 0$ . En fait,  $(n-1)^2p^3 - (n-1)p^2 + (n-2)^2 \geq (n-1)^2p^3 - (n-1)p^2 = (n-1)p^2((n-1)p-1) \geq 0$  car  $n \geq 2$  et  $p \geq 1$ .
- Si  $x \in I_{(n,p)}^3 = \left[ \frac{1}{(p+1)n-1}, \frac{p^2}{np^3-p^2+p+n-2} \right]$  pour des entiers  $n \geq 2$  et  $p \geq 1$  alors  $W_n(x) = \frac{p(x+p-1)}{np^2-p-n+2} \geq W_n\left(\frac{1}{(p+1)n-1}\right) = V_n(x)$  car  $W_n$  est strictement croissante sur  $I_{(n,p)}^3$ . De plus,  $W_n(x) = \frac{p(x+p-1)}{np^2-p-n+2} \leq 1/n$  si et seulement si  $x \leq \frac{(n-1)p-n+2}{np}$ . Par hypothèse,  $x < \frac{p^2}{np^3-p^2+p+n-2} \leq \frac{(n-1)p-n+2}{np}$  si et seulement si  $n(n-1)p^4 - (n+1)(n-1)p^3 + (2n-3)p^2 + (n-2)^2p - (n-2)^2 \geq 0$ . Si  $p = 1$  alors la dernière inégalité est équivalente à  $n \geq 2$  qui est vrai par hypothèse. Sinon,  $n(n-1)p^4 - (n+1)(n-1)p^3 + (2n-3)p^2 + (n-2)^2p - (n-2)^2 \geq n(n-1)p^4 - (n+1)(n-1)p^3 = (n-1)p^3(np - n - 1) \geq 0$  car  $n \geq 2$  et  $p \geq 2$ . ■

## B.2 Preuve de la propriété 4.2

**Propriété 4.2.** (page 103) *Étant donné un entier  $n \geq 2$ , si  $x \leq \frac{p+1}{p((p+1)n-1)}$  pour un certain entier  $p \geq 1$ , alors  $W_n(x) \geq \frac{p}{(p+1)n-1}$ .*

**Preuve.** Considérons les différentes valeurs de  $x \leq \frac{p+1}{p((p+1)n-1)}$ .

- Si  $x = \frac{p+1}{p((p+1)n-1)}$  alors  $W_n(x) = 1 - p(n-1) \frac{p+1}{p((p+1)n-1)} = \frac{p}{(p+1)n-1}$ .
- Si  $x \in I_{(n,p')}^2 \cup I_{(n,p')}^3 = \left[ \frac{1}{(p'+1)n-1}, \frac{p'+1}{p'((p'+1)n-1)} \right]$  pour des  $p' \geq p$  alors selon la propriété 4.1,  $W_n(x) \geq V_n(x) = \frac{p'}{(p'+1)n-1} \geq \frac{p}{(p+1)n-1}$  car  $p' \geq p$ .
- Si  $x \in I_{(n,p')}^1 = \left[ \frac{p'+1}{p'((p'+1)n-1)}, \frac{1}{p'n-1} \right]$  pour des  $p' \geq p+1$  alors  $W_n(x) = 1 - p'(n-1)x \geq W_n\left(\frac{1}{p'n-1}\right) = \frac{p'-1}{p'n-1}$  car  $W_n$  est strictement décroissante sur  $I_{(n,p')}^1$ . Finalement,  $\frac{p'-1}{p'n-1} \geq \frac{p}{(p+1)n-1}$  car  $p' \geq p+1$ . ■

## B.3 Preuve de la propriété 4.3

**Propriété 4.3.** (page 103) *Étant donné un entier  $n \geq 2$ , si  $x \in I_{(n,p)}$  pour un certain entier  $p \geq 1$ , alors  $(p-1)x < W_n(x) \leq px$ .*

**Preuve.** Considérons les différentes valeurs de  $x \in I_{(n,p)} = I_{(n,p)}^1 \cup I_{(n,p)}^2 \cup I_{(n,p)}^3$ .

- Si  $x \in I_{(n,p)}^1 = \left[ \frac{p+1}{p((p+1)n-1)}, \frac{1}{pn-1} \right]$  alors  $W_n(x) = 1 - p(n-1)x \leq px$  si et seulement si  $x \geq \frac{1}{np}$ . Par hypothèse,  $x \geq \frac{p+1}{p((p+1)n-1)} > \frac{1}{np}$  car  $p \geq 1$ . De plus,  $(p-1)x < W_n(x) = 1 - p(n-1)x$  car  $x < \frac{1}{pn-1}$ .
- Si  $x \in I_{(n,p)}^2 = \left[ \frac{p^2}{np^3-p^2+p+n-2}, \frac{p+1}{p((p+1)n-1)} \right]$  alors  $W_n(x) = \frac{p(1-px)}{(n-1)p+n-2} \leq px$  si et seulement si  $x \geq \frac{1}{(p+1)n-2}$ . Par hypothèse,  $x \geq \frac{p^2}{np^3-p^2+p+n-2} \geq \frac{1}{(p+1)n-2}$  si et seulement si  $(p-1)((n-1)p+n-2) \geq 0$  qui est vrai car  $n \geq 2$  et  $p \geq 1$ . De plus,  $(p-1)x < W_n(x) = \frac{p(1-px)}{(n-1)p+n-2}$  si et seulement si  $x < \frac{p}{np^2-p-n+2}$ . Par hypothèse,  $x < \frac{p+1}{p((p+1)n-1)} \leq \frac{p}{np^2-p-n+2}$  si et seulement si  $(n-1)p+n-2 \geq 0$  qui est vrai car  $n \geq 2$  et  $p \geq 1$ .
- Si  $x \in I_{(n,p)}^3 = \left[ \frac{1}{(p+1)n-1}, \frac{p^2}{np^3-p^2+p+n-2} \right]$  alors  $W_n(x) = \frac{p(x+p-1)}{np^2-p-n+2}$ . Si  $p = 1$  alors  $W_n(x) = x$  et  $0 < W_n(x) \leq x$ . Supposons que  $p \geq 2$ , alors  $\frac{p(x+p-1)}{np^2-p-n+2} \leq px$  si et seulement si  $x \geq \frac{p-1}{np^2-p-(n-1)} = \frac{1}{(p+1)n-1}$  car  $p \geq 2$  et par hypothèse  $x \geq \frac{1}{(p+1)n-1}$ . De plus,  $(p-1)x < W_n(x) = \frac{p(x+p-1)}{np^2-p-n+2}$  ssi  $x < \frac{p(p-1)}{np^3-(n+1)p^2-(n-2)p+n-2}$ . Par hypothèse,  $x < \frac{p^2}{np^3-p^2+p+n-2} \leq \frac{p(p-1)}{np^3-(n+1)p^2-(n-2)p+n-2}$  ssi  $np^2 - p - n + 2 \geq 0$  qui est vrai car  $n, p \geq 2$ . ■

## La garantie de SEUIL est au plus près

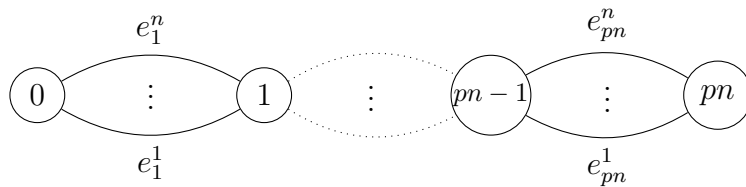
Nous présentons une famille d'instances critiques (*tight instances*) pour lesquelles la solution produite par l'algorithme SEUIL (algorithme 4.1, page 104) est au plus près (*tight analysis*). Cela signifie que l'utilité de l'agent le plus défavorisé pour la solution retournée par SEUIL est la même que celle fournie par le théorème 4.1 (page 106).

La famille d'instances provient du matroïde graphique où une base correspond à un arbre couvrant. Ces instances vérifient  $\alpha_1 = \dots = \alpha_n = \alpha$ . On dira qu'une instance est bien définie si pour toute arête  $e$  du graphe correspondant,  $\alpha \geq u_i(e) \geq 0$ . Ces instances sont toutes normalisées à 1. Nous voulons alors montrer que pour tout  $\alpha \in [0, 1]$ , la solution  $B$  retournée par SEUIL vérifie pour ces instances

$$\min_{i \in [n]} u_i(B) = \min_{i \in [n]} \{W_n(\alpha) u_i(B_i^*)\} = W_n(\alpha)$$

Lorsque  $\alpha \in [\frac{1}{n-1}, 1]$ , nous avons vu qu'il n'y a aucune garantie c'est-à-dire  $W_n(\alpha) = 0$ . Nous présentons les instances pour lesquelles la garantie mentionnée est atteinte en fonction des valeurs de  $\alpha \in [0, \frac{1}{n-1}]$ .

1. Si  $\alpha \in I_{(n,p)}^1 = [\frac{p+1}{p((p+1)n-1)}, \frac{1}{pn-1}]$  pour des entiers  $n \geq 2$  et  $p \geq 1$  alors SEUIL fournit  $u_i(B) \geq W_n(\alpha) u_i(B_i^*) = 1 - p(n-1)\alpha$  (car les instances sont normalisées). Montrons dans l'instance suivante que l'utilité de l'agent le moins heureux coïncide avec la garantie de SEUIL, c'est-à-dire  $\min_{i \in N} u_i(B) = W_n(\alpha)$ .



Nous avons ici un multigraphe  $G = (V, E)$  tel que  $V = \{0, 1, \dots, pn\}$  est l'ensemble des sommets et  $E = \bigcup_{k=1}^{pn} \{e_k^1, \dots, e_k^n\}$  est l'ensemble des arêtes. Il y a  $n$  arêtes  $e_k^1, \dots, e_k^n$  entre chaque paire de sommets consécutifs  $(k-1, k)$

pour  $k \in \{1, \dots, pn - 1\}$ . Les utilités sont : pour tout  $i, j \in \{1, \dots, n\}$  : pour tout  $k \in \{1, \dots, pn - 1\}$ ,  $u_i(e_k^j) = \alpha$  si  $i = j$  et  $u_i(e_k^j) = 0$  sinon. De plus,  $u_i(e_{pn}^j) = 1 - (pn - 1)\alpha$  si  $i = j$  et  $u_i(e_{pn}^j) = 0$  sinon. L'instance est bien définie pour les raisons suivantes :

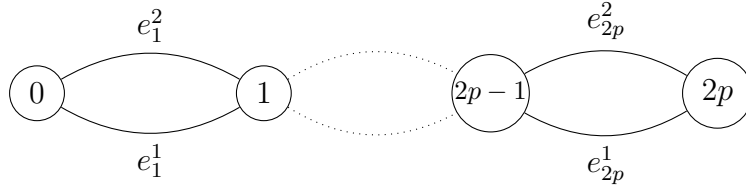
- $\alpha \geq \frac{p+1}{p((p+1)n-1)} > 0$  car  $n \geq 2$  et  $p \geq 1$  ;
- $1 - (pn - 1)\alpha > 0$  car  $\alpha < \frac{1}{pn-1}$  ;
- $\alpha \geq 1 - (pn - 1)\alpha$  si et seulement si  $\alpha \geq \frac{1}{pn}$ . Par hypothèse  $\alpha \geq \frac{p+1}{p((p+1)n-1)}$ , donc  $\alpha \geq \frac{p+1}{p((p+1)n-1)} \geq \frac{1}{pn}$  est toujours satisfaite.

Soit  $B_i^* = \{e_1^i, \dots, e_{pn}^i\}$  un arbre couvrant optimal pour l'agent  $i \in \{1, \dots, n\}$  tel que  $u_i(B_i^*) = \sum_{j=1}^{pn} u_i(e_j^i) = (pn - 1)\alpha + 1 - (pn - 1)\alpha = 1$ . Alors l'instance est normalisée.

L'algorithme SEUIL retourne la base  $B = \cup_{i=1}^n \{e_{(i-1)p+1}^i, e_{(i-1)p+2}^i, \dots, e_{ip}^i\}$ . L'instance est normalisée, donc  $\tilde{u}_i(B) = u_i(B) = p\alpha \geq W_n(\alpha)$  en utilisant la propriété 4.3 (page 103) pour  $i = 1, \dots, n - 1$  et  $\tilde{u}_n(B) = u_n(B) = (p - 1)\alpha + 1 - (pn - 1)\alpha = 1 - p(n - 1)\alpha = W_n(\alpha)$ . Alors l'utilité de l'agent le moins heureux est  $\min_{i \in N} u_i(B) = \min\{p\alpha, 1 - p(n - 1)\alpha\} = 1 - p(n - 1)\alpha$  si et seulement si  $\alpha \geq \frac{1}{pn}$  et par hypothèse,  $\alpha \geq \frac{p+1}{p((p+1)n-1)} \geq \frac{1}{pn}$ . Ainsi,  $\min_{i \in N} u_i(B) = 1 - p(n - 1)\alpha = W_n(\alpha)$ .

2. Soit  $\alpha \in I_{(n,p)}^2 = \left[ \frac{p^2}{np^3 - p^2 + p + n - 2}, \frac{p+1}{p((p+1)n-1)} \right]$  pour des entiers  $n \geq 2$  et  $p \geq 1$ . Nous avons deux instances différentes selon que  $n = 2$  ou  $n \geq 3$ .

(a) Si  $n = 2$  alors considérons l'instance suivante :



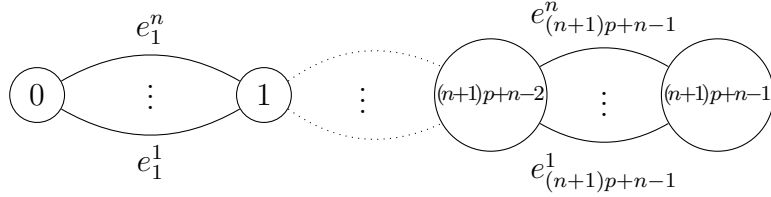
Pour  $i, j = 1, 2$  : pour tout  $k \in \{1, \dots, 2p - 1\}$ ,  $u_i(e_k^j) = \alpha$  si  $i = j$  et  $u_i(e_k^j) = 0$  sinon. De plus,  $u_i(e_{2p}^j) = 1 - (2p - 1)\alpha$  si  $i = j$  et  $u_i(e_{2p}^j) = 0$  sinon. L'instance est bien définie pour les raisons suivantes :

- $\alpha \geq \frac{p}{2p^2 - p + 1} > 0$  car  $p \geq 1$  ;
- $1 - (2p - 1)\alpha > 0$  si et seulement si  $\alpha < \frac{1}{2p-1}$ . Puisque  $\alpha \in I_{(2,p)}^2$ , alors  $\alpha < \frac{p+1}{p(2p+1)} < \frac{1}{2p-1}$  est toujours vrai.
- $\alpha \geq 1 - (2p - 1)\alpha$  si et seulement si  $\alpha \geq \frac{1}{2p}$ . Par hypothèse  $\alpha \geq \frac{p}{2p^2 - p + 1}$ , donc  $\alpha \geq \frac{p}{2p^2 - p + 1} \geq \frac{1}{2p}$  car  $p \geq 1$ .

Soit  $B_i^* = \{e_1^i, \dots, e_{2p}^i\}$  un arbre couvrant optimal pour l'agent  $i \in \{1, 2\}$  tel que  $u_i(B_i^*) = \sum_{j=1}^{2p} u_i(e_j^i) = (2p - 1)\alpha + 1 - (2p - 1)\alpha = 1$ . Alors l'instance est normalisée.

SEUIL retourne l'arbre couvrant  $B = \{e_1^1, \dots, e_p^1\} \cup \{e_{p+1}^2, \dots, e_{2p}^2\}$  tel que  $\tilde{u}_1(B) = u_1(B) = p\alpha \geq W_2(\alpha)$  en utilisant la propriété 4.3 (page 103) et  $\tilde{u}_2(B) = u_2(B) = (p-1)\alpha + 1 - (2p-1)\alpha = 1 - p\alpha = W_2(\alpha)$ . Alors l'utilité de l'agent le moins heureux est  $\min_{i \in N} u_i(B) = \min\{p\alpha, 1 - p\alpha\} = 1 - p\alpha = W_2(\alpha)$ .

(b) Considérons l'instance suivante pour  $n \geq 3$  :



Pour tout  $i, j \in \{1, \dots, n\}$  : pour tout  $k \in \{1, \dots, p\}$ ,  $u_i(e_k^j) = \alpha$  si  $i = j$  et  $u_i(e_k^j) = 0$  sinon ; pour tout  $k \in \{p+1, \dots, (n+1)p+n-1\}$ ,  $u_i(e_k^j) = \frac{1-p\alpha}{(p+1)n-1}$  si  $i = j$  et  $u_i(e_k^j) = 0$  sinon. L'instance est bien définie pour les raisons suivantes :

- $\alpha \geq \frac{p^2}{np^3 - p^2 + p + n - 2} > 0$  car  $n \geq 3$  et  $p \geq 1$  ;
- $\frac{1-p\alpha}{(p+1)n-1} \geq 0$  car  $(p+1)n-1 > 0$  (puisque  $n \geq 3$  et  $p \geq 1$ ) et  $1 - p\alpha \geq 0$  si et seulement si  $\alpha \leq \frac{1}{p}$ . Par hypothèse  $\alpha < \frac{p+1}{p((p+1)n-1)}$ , donc  $\frac{p+1}{p((p+1)n-1)} \leq \frac{1}{p}$  si et seulement si  $p \geq \frac{1}{n-1} - 1$  qui est vrai puisque  $n \geq 3$  et  $p \geq 1$  ;
- $\alpha \geq \frac{1-p\alpha}{np+n-1}$  si et seulement si  $\alpha \geq \frac{1}{(n+1)p+n-1}$ . Par hypothèse  $\alpha \geq \frac{p^2}{np^3 - p^2 + p + n - 2}$ , donc  $\alpha \geq \frac{p^2}{np^3 - p^2 + p + n - 2} \geq \frac{1}{(n+1)p+n-1}$  si et seulement si  $p^3 + np^2 - p - n + 2 \geq 0$  qui est vérifié car  $n \geq 3$  et  $p \geq 1$ .

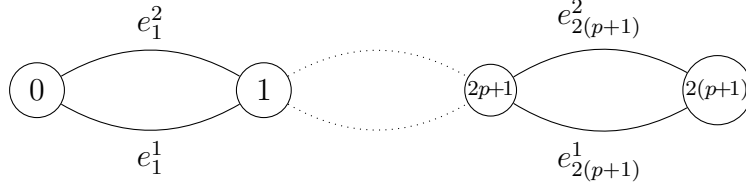
Pour tout  $i \in N$ , l'arbre couvrant  $B_i^* = \{e_1^i, \dots, e_{(n+1)p+n-1}^i\}$  est optimal pour l'agent  $i$  avec  $u_i(B_i^*) = \sum_{j=1}^{(n+1)p+n-1} u_i(e_j^i) = p\alpha + (np+n-1) \frac{1-p\alpha}{np+n-1} = 1$ . Alors l'instance est normalisée.

L'application de SEUIL donne l'arbre couvrant  $B = \cup_{i \in N} B_i$  tel que  $B_1 = \{e_1^1, \dots, e_p^1\}$ ,  $B_i = \{e_{(i-1)p+1}^i, \dots, e_{ip}^i\}$  pour tout  $i \in \{2, \dots, n-1\}$  et  $B_n = \{e_{(n-1)p+1}^n, \dots, e_{(n+1)p+n-1}^n\}$ . L'instance est normalisée, donc  $\tilde{u}_1(B) = u_1(B) = u_1(B_1) = p\alpha \geq W_n(\alpha)$  en utilisant la propriété 4.3 (page 103),  $\tilde{u}_i(B) = u_i(B) = u_i(B_i) = p \frac{1-p\alpha}{(p+1)n-1} = W_n(\alpha)$  pour tout  $i \in \{2, \dots, n-1\}$  car  $\alpha \in I_{(n,p)}^2$ , et  $\tilde{u}_n(B) = u_n(B) = u_n(B_n) = (2p+n-1) \frac{1-p\alpha}{(p+1)n-1} > \frac{p(1-p\alpha)}{(p+1)n-1} = W_n(\alpha)$ . D'où,  $\min_{i \in N} u_i(B) = \min\{p\alpha, \frac{p(1-p\alpha)}{(p+1)n-1}, (2p+n-1) \frac{1-p\alpha}{(p+1)n-1}\} = \frac{p(1-p\alpha)}{(p+1)n-1} = W_n(\alpha)$ .

3. Soit  $\alpha \in I_{(n,p)}^3 = \left[ \frac{1}{(p+1)n-1}, \frac{p^2}{np^3 - p^2 + p + n - 2} \right]$  pour des entiers  $n \geq 2$  et  $p \geq 1$ . Nous avons deux instances différentes selon que  $n = 2$  ou  $n \geq 3$ .



(a) Si  $n = 2$  alors considérons l'instance suivante du matroïde graphique :



Les utilités des arêtes sont : pour  $i, j = 1, 2$  :  $u_i(e_1^j) = \alpha$  si  $i = j$  et  $u_i(e_1^j) = 0$  sinon ;  $u_i(e_{2(p+1)}^j) = \frac{(2p+1)\alpha-1}{2p-1}$  si  $i = j$  et  $u_i(e_{2(p+1)}^j) = 0$  sinon ; et pour tout  $k \in \{2, \dots, 2p+1\}$ ,  $u_i(e_k^j) = \frac{1-2\alpha}{2p-1}$  si  $i = j$  et  $u_i(e_k^j) = 0$  sinon. L'instance est bien définie pour les raisons suivantes :

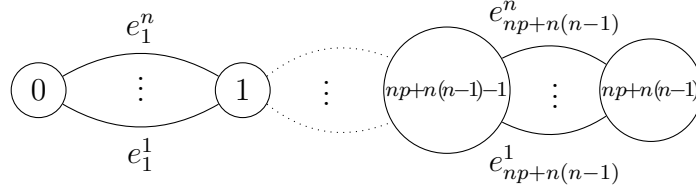
- $\alpha \geq \frac{1}{2p+1} > 0$  car  $p \geq 2$  ;
- $\frac{(2p+1)\alpha-1}{2p-1} \geq 0$  car  $2p-1 > 0$  (puisque  $p \geq 1$ ) et  $(2p+1)\alpha-1 \geq 0$  car  $\alpha \geq \frac{1}{2p+1}$  ;
- $\frac{1-2\alpha}{2p-1} \geq 0$  car  $2p-1 > 0$  et  $1-2\alpha \geq 0$  si et seulement si  $\alpha \leq \frac{1}{2}$  et par hypothèse  $\alpha < \frac{p}{2p^2-p+1}$ , donc  $\alpha < \frac{p}{2p^2-p+1} \leq \frac{1}{2}$  car  $p \geq 1$  ;
- $\alpha \geq \frac{(2p+1)\alpha-1}{2p-1}$  si et seulement si  $\alpha \leq \frac{1}{2}$  qui est vrai comme montré précédemment ;
- $\alpha \geq \frac{1-2\alpha}{2p-1}$  car  $\alpha \geq \frac{1}{2p+1}$ .

Soit  $B_i^* = \{e_1^i, \dots, e_{2(p+1)}^i\}$  un arbre couvrant optimal pour l'agent  $i \in \{1, 2\}$  tel que  $u_i(B_i^*) = \sum_{j=1}^{2(p+1)} u_i(e_j^i) = \alpha + \frac{(2p+1)\alpha-1}{2p-1} + 2p\frac{1-2\alpha}{2p-1} = 1$ . Alors l'instance est normalisée.

L'application de SEUIL fournit l'arbre couvrant  $B = B_1 \cup B_2$  tel que  $B_1 = \{e_1^1, \dots, e_p^1\}$  et  $B_2 = \{e_{p+1}^2, \dots, e_{2(p+1)}^2\}$ . Puisque l'instance est normalisée, alors  $\tilde{u}_1(B) = u_1(B) = u_1(B_1) = \alpha + \frac{(p-1)(1-2\alpha)}{2p-1} = \frac{\alpha+p-1}{2p-1} = W_2(\alpha)$  et  $\tilde{u}_2(B) = u_2(B) = u_2(B_2) = \frac{(p+1)(1-2\alpha)}{2p-1} + \frac{(2p+1)\alpha-1}{2p-1} = \frac{p-\alpha}{2p-1} \geq \frac{\alpha+p-1}{2p-1} = W_2(\alpha)$  si et seulement si  $\alpha \leq 1/2$  et par hypothèse  $\alpha < \frac{p}{2p^2-p+1} \leq \frac{1}{2}$  car  $p \geq 1$ . D'où,  $\min\{u_1(B), u_2(B)\} = \min\left\{\frac{\alpha+p-1}{2p-1}, \frac{p-\alpha}{2p-1}\right\} = \frac{\alpha+p-1}{2p-1} = W_2(\alpha)$ .

(b) Si  $n \geq 3$  alors considérons l'instance suivante :

Pour tout  $i, j \in \{1, \dots, n\}$  :  $u_i(e_1^j) = \alpha$  si  $i = j$  et  $u_i(e_1^j) = 0$  sinon ; pour tout  $k \in \{2, \dots, p\}$ ,  $u_i(e_k^j) = \frac{p-\alpha(np+n-2)}{np^2-p-n+2}$  si  $i = j$  et  $u_i(e_k^j) = 0$  sinon ; et pour tout  $k \in \{p+1, \dots, np+n(n-1)\}$ ,  $u_i(e_k^j) = \frac{(n-1)p^2-n+2-p\alpha}{(n-1)(p+n)(np^2-p-n+2)}$  si  $i = j$  et  $u_i(e_k^j) = 0$  sinon. L'instance est bien définie pour les raisons suivantes :



- $\alpha \geq \frac{1}{(p+1)n-1} > 0$  car  $n \geq 3$  et  $p \geq 1$ .
- $\frac{p-\alpha(np+n-2)}{np^2-p-n+2} \geq 0$  car  $np^2 - p - n + 2 \geq 0$  (puisque  $n \geq 3$  et  $p \geq 1$ ) et  $p - \alpha(np + n - 2) \geq 0$  si et seulement si  $\alpha \leq \frac{p}{np+n-2}$ . Par hypothèse  $\alpha < \frac{p^2}{np^3-p^2+p+n-2}$ , donc  $\alpha < \frac{p^2}{np^3-p^2+p+n-2} \leq \frac{p}{np+n-2}$  si et seulement si  $np^3 - (n+1)p^2 - (n-3)p + n - 2 \geq 0$  qui est vrai car  $p \geq 1$  et  $n \geq 3$ .
- $\frac{(n-1)p^2-n+2-p\alpha}{(n-1)(p+n)(np^2-p-n+2)} \geq 0$  car  $(n-1)(p+n)(np^2-p-n+2) > 0$  (puisque  $n \geq 3$  et  $p \geq 1$ ) et  $(n-1)p^2 - n + 2 - p\alpha \geq 0$  si et seulement si  $\alpha \leq \frac{(n-1)p^2-n+2}{p}$ . Par hypothèse  $\alpha < \frac{p^2}{np^3-p^2+p+n-2} \leq \frac{(n-1)p^2-n+2}{p}$  si et seulement si  $p^3((n-1)(np-1)p-n^2+3n-2)+(n-2)(np^2-p-n+2) \geq 0$  qui est vrai car  $n \geq 3$  et  $p \geq 1$ .
- $\alpha \geq \frac{p-\alpha(np+n-2)}{np^2-p-n+2}$  car  $\alpha \geq \frac{1}{(p+1)n-1}$ .
- $\alpha \geq \frac{(n-1)p^2-n+2-p\alpha}{(n-1)(p+n)(np^2-p-n+2)}$  si et seulement si  $\alpha \geq \frac{(n-1)p^2-n+2}{(n-1)(p+n)(np^2-p-n+2)+p}$  et par hypothèse  $\alpha \geq \frac{1}{(p+1)n-1} \geq \frac{(n-1)p^2-n+2}{(n-1)(p+n)(np^2-p-n+2)+p}$  si et seulement si  $np^2 - p - n + 2 \geq 0$  qui est vrai car  $n \geq 3$  et  $p \geq 1$ .

L'arbre couvrant  $B_i^* = \{e_1^i, \dots, e_{np+n(n-1)}^i\}$  est optimal pour tout  $i \in N$  avec  $u_i(B_i^*) = \sum_{j=1}^{np+n(n-1)} u_i(e_j^i) = \alpha + (p-1)\frac{p-\alpha(np+n-2)}{np^2-p-n+2} + (n-1)(p+n)\frac{(n-1)p^2-n+2-p\alpha}{(n-1)(p+n)(np^2-p-n+2)} = 1$ . Alors l'instance est normalisée.

SEUIL produit la solution  $B = \cup_{i \in N} B_i$  où  $B_1 = \{e_1^1, \dots, e_p^1\}$ ,  $B_i = \{e_{(i-1)p+(i-2)(n-1)+1}^i, \dots, e_{ip+(i-1)(n-1)}^i\}$ ,  $\forall i \in \{2, \dots, n-1\}$  et  $B_n = \{e_{(n-1)p+(n-2)(n-1)+1}^n, \dots, e_{np+n(n-1)}^n\}$ .

L'instance est normalisée, donc  $\tilde{u}_1(B) = u_1(B) = u_1(B_1) = \alpha + (p-1)\frac{p-\alpha(np+n-2)}{np^2-p-n+2} = \frac{p(\alpha+p-1)}{np^2-p-n+2} = W_n(\alpha)$ . Pour tout  $i \in \{2, \dots, n-1\}$ ,  $\tilde{u}_i(B) = u_i(B) = u_i(B_i) = \frac{(p+n-1)((n-1)p^2-n+2-p\alpha)}{(n-1)(p+n)(np^2-p-n+2)} \geq \frac{p(\alpha+p-1)}{np^2-p-n+2} = W_n(\alpha)$  si et seulement si  $\alpha \leq \frac{(n^2-2n+2)p-(n-1)(n-2)}{p(np+(n-1)(n+1))}$  et par hypothèse

$\alpha < \frac{p^2}{np^3-p^2+p+n-2} \leq \frac{(n^2-2n+2)p-(n-1)(n-2)}{p(np+(n-1)(n+1))}$  si et seulement si  $(n^3 - 2n^2 + n)p^4 - (n^3 - n^2 + 1)p^3 + (2n^2 - 5n + 4)p^2 + (n^3 - 5n^2 + 9n - 6)p - (n-1)(n-2)^2 \geq 0$  qui est vrai car  $n \geq 3$  et  $p \geq 1$ . De plus,  $\tilde{u}_n(B) = u_n(B) = u_n(B_n) = (p+2(n-1))\frac{(n-1)p^2-n+2-p\alpha}{(n-1)(p+n)(np^2-p-n+2)} > (p+n-$

$$1) \frac{(n-1)p^2-n+2-p\alpha}{(n-1)(p+n)(np^2-p-n+2)} \geq \frac{p(\alpha+p-1)}{np^2-p-n+2} = W_n(\alpha).$$

$$\text{D'où, } \min_{i \in N} u_i(B) = \min \left\{ \frac{p(\alpha+p-1)}{np^2-p-n+2}, (p+n-1) \frac{(n-1)p^2-n+2-p\alpha}{(n-1)(p+n)(np^2-p-n+2)}, (p+2(n-1)) \frac{(n-1)p^2-n+2-p\alpha}{(n-1)(p+n)(np^2-p-n+2)} \right\} = \frac{p(\alpha+p-1)}{np^2-p-n+2} = W_n(\alpha).$$

En conclusion, l'analyse de SEUIL est au plus près (*tight*) pour tout  $\alpha \in [0, 1]$ .

## Preuves des lemmes 5.1 et 5.2

### D.1 Preuve du lemme 5.1

**Lemme 5.1.** (page 133) *Si  $n = |N| \leq 9$  alors le protocole 5.3 est bien défini.*

**Preuve.** L'arborescence de la preuve est proposée dans la figure 5.1 (page 132).

Pour  $n \leq 2$ , le résultat est trivial. Supposons que  $n \geq 3$  et montrons que le protocole 5.3 (page 130) aboutit à l'instruction **retourner** dans tous les cas.

L'ensemble  $\{k \geq 1 : |N_k| \geq n - k - 2\}$  est non vide car  $|N_{n-2}| \geq 0$ , donc  $k_0$  existe et  $1 \leq k_0 \leq n - 2$ . De plus,  $|N_k| \leq n - 2 \leq 7$  car  $|N_k| \leq |N \setminus \{i_1, i_2\}|$  et  $|N| \leq 9$ .

Si  $k_0 = 1$  alors  $|N_{k_0}| \leq n - 2 = n - k_0 - 1$ . Donc nous sommes aux étapes 6-9 du protocole et l'instruction **retourner** est atteinte (étape 9). Supposons maintenant que  $k_0 \geq 2$ . Nous pouvons également supposer que  $|N_{k_0}| \geq n - k_0$  puisqu'autrement nous serions dans les étapes 6-9.

Si  $k_0 = 2$  alors  $n - 2 = n - k_0 \leq |N_{k_0}| \leq n - 2$  et forcément  $|N_{k_0-1} \setminus N_{k_0}| = 0$ , donc nous sommes aux étapes 12-16 où l'instruction **retourner** est atteinte à l'étape 16. Maintenant, supposons que  $k_0 \geq 3$ . Alors  $N_{k_0-1}$  et  $N_{k_0-2}$  existent.

S'il existe un certain  $p \in \{1, 2\}$  tel que  $|N_{k_0-p} \setminus N_{k_0}| = p$  ou  $|N_{k_0-p} \setminus N_{k_0}| = p - 1$  alors nous sommes aux étapes 12-16 du protocole où l'instruction **retourner** apparaît à l'étape 16. Supposons dorénavant que  $|N_{k_0-1} \setminus N_{k_0}| \geq 2$  et soit  $|N_{k_0-2} \setminus N_{k_0}| = 0$  ou  $|N_{k_0-2} \setminus N_{k_0}| \geq 3$ .

Si  $|N_{k_0-2} \setminus N_{k_0}| = 0$  alors nous sommes aux étapes 17-21 du protocole où l'instruction **retourner** apparaît à l'étape 21. Supposons que  $|N_{k_0-2} \setminus N_{k_0}| \geq 3$ . Alors

$$(n - k_0) + 3 \leq |N_{k_0}| + |N_{k_0-2} \setminus N_{k_0}| \leq |N \setminus \{i_1, i_2\}| = n - 2$$

Ainsi,  $k_0 \geq 5$  et  $N_{k_0-3}$  existe.

Si  $|N_{k_0-3} \setminus N_{k_0}| \leq 2$  alors nous sommes aux étapes 22-27 du protocole où l'instruction **retourner** apparaît à l'étape 27. Supposons que  $|N_{k_0-3} \setminus N_{k_0}| \geq 3$ . Tous les autres cas sont dans les étapes 28-31 du protocole où l'instruction **retourner** apparaît à l'étape 31.

Aux étapes 28-31, montrons que  $|N_{k_0-3} \setminus N_{k_0}| = 3$ , c'est-à-dire que le cas  $|N_{k_0-3} \setminus N_{k_0}| \geq 4$  ne se produira jamais. Par contradiction, supposons que  $|N_{k_0-3} \setminus N_{k_0}| \geq 4$ . Alors

$$4 \leq |N_{k_0-3} \setminus N_{k_0}| \leq |N_{k_0-3}| < n - (k_0 - 3) - 2 \leq 9 - (k_0 - 3) - 2 \quad (\text{D.1})$$

(voir l'étape 5), ainsi  $k_0 < 6$ . De plus,

$$(n - k_0) + 4 \leq |N_{k_0}| + |N_{k_0-3} \setminus N_{k_0}| \leq |N \setminus \{i_1, i_2\}| = n - 2$$

D'où,  $k_0 \geq 6$ , contradiction avec (D.1).

Dans tous les cas, le résultat est prouvé. Donc, le protocole 5.3 s'achève toujours car, dans tous les cas, il fait des appels récursifs sur des instances plus petites. ■

## D.2 Preuve du lemme 5.2

**Lemme 5.2.** (page 135) *Si  $n \leq 9$  alors à chaque appel récursif du protocole 5.3, un ensemble  $I'$  d'agents et un ensemble indépendant  $B_{I'} \in \mathcal{F}$  sont obtenus tels que  $OPT_i(\mathcal{M}/B_{I'}) \geq \frac{|I'|}{n} OPT_i(\mathcal{M})$  pour tout  $i \in I' \setminus \{i_1\}$ .*

**Preuve.** Nous faisons une étude de cas. Si  $n = 1$  (correspondant à l'étape 1 du protocole 5.3, page 130) alors cela est vrai en posant  $I' = \{i_1\}$  et  $B_{I'} = \emptyset$ . Si  $n = 2$  (correspondant à l'étape 3) alors en utilisant l'hypothèse 5.1 (page 135) et en posant  $I' = \{i_2\}$ ,  $B_{I'} = A_{\sigma(1)}$ , on obtient  $OPT_{i_2}(\mathcal{M}/B_{I'}) \geq \frac{1}{2} OPT_{i_2}(\mathcal{M}) = \frac{|I'|}{n} OPT_{i_2}(\mathcal{M})$  (voir la section 5.2, page 126).

Maintenant, supposons que  $n \geq 3$ . Nous suivons les étapes de la preuve du lemme 5.1 (page 187). Nous savons que  $k_0$  existe et  $k_0 \leq n - 2$ . Si  $|N_{k_0}| \leq n - k_0 - 1$  (correspondant aux étapes 6-9), alors soit  $|N_{k_0}| = n - k_0 - 2$  ou  $|N_{k_0}| = n - k_0 - 1$ . Dans tous les cas,  $|J_0| = n - k_0$ . Posons  $I' = J_0$  et  $B_{I'} = \cup_{j \leq k_0} A_{\sigma(j)}$ . En utilisant l'hypothèse 5.1 (page 135), on obtient  $\forall i \in I' \setminus \{i_1\}$ ,  $OPT_i(\mathcal{M}/B_{I'}) \geq \frac{|I'|}{n} OPT_i(\mathcal{M})$  (correspondant à l'étape 8) car  $|I'| = |J_0| = n - k_0$ .

À l'étape 9 et par construction,  $\forall i \in N \setminus (J_0 \cup \{i_1\})$ ,  $OPT_i(\mathcal{M}/(\cup_{j \leq k_0} A_{\sigma(j)})) < \frac{|J_0|}{n} OPT_i(\mathcal{M})$  car  $|J_0| = n - k_0$  (voir l'observation 5.1, page 135). En utilisant le lemme 1.1 (page 32) avec  $\mathcal{M}$ ,  $S = \cup_{j \leq k_0} A_{\sigma(j)}$  et  $B_{I'} = B_{J_0}$  où  $B_{J_0}$  est la base obtenue à l'étape 8, on obtient  $\forall i \in I' = N \setminus J_0$  avec  $i \neq i_1$ ,  $OPT_i(\mathcal{M}/B_{I'}) \geq \frac{|N \setminus J_0|}{n} OPT_i(\mathcal{M}) = \frac{|I'|}{n} OPT_i(\mathcal{M})$  car  $OPT_i(\mathcal{M}/S) < \frac{|J_0|}{n} OPT_i(\mathcal{M})$ .

Aux étapes 10-11, nous savons par construction que  $|N_{k_0}| \geq n - k_0$  (puisque autrement nous serions à l'étape 6). Afin de construire  $J_1$ , nous commençons par ajouter  $\{i_2\}$  puis on ajoute de nouveaux éléments de  $N_{k_0} \cap N_{k_0-1}$ ,  $N_{k_0} \cap N_{k_0-2}$  et ainsi de suite jusqu'à  $N_{k_0} \cap N_0$ . L'ensemble  $J_1$  est bien défini et sa taille est  $n - k_0$

car  $N_{k_0} \cap N_0 = N_{k_0} \subseteq N_0$ . Soit  $I' = J_1$  et  $B_{I'} = \cup_{j \leq k_0} A_{\sigma(j)}$ . Par construction, on a  $\forall i \in I'$ ,  $OPT_i(\mathcal{M}/B_{I'}) \geq \frac{n-k_0}{n} OPT_i(\mathcal{M}) = \frac{|I'|}{n} OPT_i(\mathcal{M})$  (correspondant à l'étape 11).

Aux étapes 12-16, soit le plus petit  $p_0 \in \{1, 2\}$  tel que  $|N_{k_0-p_0} \setminus N_{k_0}| = p_0$  ou  $|N_{k_0-p_0} \setminus N_{k_0}| = p_0 - 1$ . Soit  $I' = J_2$  où  $J_2$  est obtenu à l'étape 14 et posons  $B_{I'} = \cup_{j \leq k_0-p_0} A_{\sigma(j)} \cup B_{J_1}$  où  $B_{J_1}$  est obtenu à l'étape 11. Utilisons le lemme 1.1 (page 32) avec  $\mathcal{M}/(\cup_{j \leq k_0-p_0} A_{\sigma(j)})$ ,  $S = \cup_{k_0-p_0+1}^{k_0} A_{\sigma(j)}$  et  $T = B_{J_1}$  afin d'obtenir  $\forall i \in I' \setminus \{i_1\}$ ,  $OPT_i(\mathcal{M}/(\cup_{j \leq k_0-p_0} A_{\sigma(j)} \cup B_{J_1})) = OPT_i(\mathcal{M}/B_{I'}) \geq \frac{p_0}{n} OPT_i(\mathcal{M}) = \frac{|I'|}{n} OPT_i(\mathcal{M})$  car en utilisant l'hypothèse 5.1,  $OPT_i(\mathcal{M}/(\cup_{j \leq k_0} A_{\sigma(j)})) < \frac{n-k_0}{n} OPT_i(\mathcal{M})$  et  $OPT_i(\mathcal{M}/(\cup_{j \leq k_0-p_0} A_{\sigma(j)})) \geq \frac{n-(k_0-p_0)}{n} OPT_i(\mathcal{M})$  puisque  $i \in N_{k_0-p_0} \setminus N_{k_0}$  (correspondant à l'étape 15).

À l'étape 16, montrons que  $N \setminus (J_1 \cup J_2) \subseteq N \setminus N_{k_0-p_0}$  ou de façon équivalente  $N_{k_0-p_0} \subseteq (J_1 \cup J_2)$ . Nous savons que  $(N_{k_0-p_0} \setminus N_{k_0}) \subseteq J_2$  grâce à la définition de  $J_2$  (voir l'étape 14), donc il reste à montrer que  $(N_{k_0-p_0} \cap N_{k_0}) \subset (J_1 \cup J_2)$  selon les valeurs de  $p_0 \in \{1, 2\}$ .

- Si  $p_0 = 1$  alors  $|N_{k_0-1} \cap N_{k_0}| \leq |N_{k_0-1}| \leq n - (k_0 - 1) - 3 = n - k_0 - 2$  (voir l'étape 5). Par définition de  $J_1$ ,  $|J_1| = n - k_0$ . En particulier,  $J_1$  contient  $i_2$  et  $N_{k_0-1} \cap N_{k_0}$  et le résultat s'ensuit (voir la construction de  $J_1$  à l'étape 10).
- Si  $p_0 = 2$  alors  $|N_{k_0-1} \setminus N_{k_0}| \geq 2$  et  $1 \leq |N_{k_0-2} \setminus N_{k_0}| \leq 2$  (voir l'étape 13). Nous voulons montrer que  $N_{k_0-2} \cap N_{k_0} \subset J_1$ , cela signifie que  $J_1$  contient au moins  $i_2$ ,  $N_{k_0-1} \cap N_{k_0}$  et  $N_{k_0-2} \cap N_{k_0}$ . Ceci est équivalent à montrer que

$$|J_1| = n - k_0 \geq |\{i_2\}| + |N_{k_0-1} \cap N_{k_0}| + |(N_{k_0-2} \cap N_{k_0}) \setminus N_{k_0-1}|$$

Par contradiction, supposons que

$$n - k_0 < 1 + |N_{k_0-1} \cap N_{k_0}| + |(N_{k_0-2} \cap N_{k_0}) \setminus N_{k_0-1}| \quad (\text{D.2})$$

D'une part,

$$|N_{k_0-1} \cap N_{k_0}| = |N_{k_0-1}| - |N_{k_0-1} \setminus N_{k_0}| \leq (n - (k_0 - 1) - 3) - 2 = n - k_0 - 4$$

car grâce à l'étape 5 nous savons que  $|N_{k_0-1}| \leq n - (k_0 - 1) - 3$  et  $|N_{k_0-1} \setminus N_{k_0}| \geq 2$  puisque  $p_0 > 1$ . D'autre part,

$$|(N_{k_0-2} \cap N_{k_0}) \setminus N_{k_0-1}| \leq |N_{k_0-2}| - |N_{k_0-2} \setminus N_{k_0}| \leq (n - (k_0 - 2) - 3) - 1 = n - k_0 - 2$$

car grâce à l'étape 5,  $|N_{k_0-2}| \leq n - (k_0 - 2) - 3$  et  $|N_{k_0-2} \setminus N_{k_0}| \geq 1$  puisque  $p_0 = 2$ . En utilisant (D.2), on obtient  $n - k_0 < 1 + (n - k_0 - 4) + (n - k_0 - 2)$ , donc

$$k_0 < n - 5 \leq 9 - 5 = 4 \quad (\text{D.3})$$

De plus,  $(n - k_0) + 2 \leq |N_{k_0}| + |N_{k_0-1} \setminus N_{k_0}| \leq |N \setminus \{i_1, i_2\}| = n - 2$ , donc  $k_0 \geq 4$ , contradiction avec (D.3). Ainsi,  $|J_1| = n - k_0 \geq 1 + |N_{k_0-1} \cap N_{k_0}| + |(N_{k_0-2} \cap N_{k_0}) \setminus N_{k_0-1}|$  et le résultat s'ensuit.

Finalement,  $N_{k_0-p_0} \subseteq (J_1 \cup J_2)$  pour tout  $p_0 \in \{1, 2\}$  et de façon équivalente,  $N \setminus (J_1 \cup J_2) \subseteq N \setminus N_{k_0-p_0}$ .

Soit  $I' = N \setminus (J_1 \cup J_2)$  et  $B_{I'} = B_{J_1} \cup B_{J_2}$ . Utilisons le lemme 1.1 (page 32) avec  $\mathcal{M}$ ,  $S = \cup_{j \leq k_0-p_0} A_{\sigma(j)}$  et  $T = B_{I'}$  pour obtenir  $\forall i \in I' \setminus \{i_1\}$ ,  $OPT_i(\mathcal{M}/B_{I'}) \geq \frac{k_0-p_0}{n} OPT_i(\mathcal{M}) = \frac{|I'|}{n} OPT_i(\mathcal{M})$  car  $OPT_i(\mathcal{M}/(\cup_{j \leq k_0-p_0} A_{\sigma(j)})) < \frac{n-(k_0-p_0)}{n} OPT_i(\mathcal{M})$  puisque  $I' \subseteq N \setminus N_{k_0-p_0}$  (correspondant à l'étape 16).

Aux étapes 17-31, nous savons que  $|N_{k_0-1} \setminus N_{k_0}| \geq 2$  et soit  $|N_{k_0-2} \setminus N_{k_0}| = 0$  ou  $|N_{k_0-2} \setminus N_{k_0}| \geq 3$ .

Si  $|N_{k_0-2} \setminus N_{k_0}| = 0$  alors nous sommes aux étapes 17-21. Soit  $I' = J_3 = \{i_3\} \subset N_{k_0-1} \setminus N_{k_0}$  (puisque  $|N_{k_0-1} \setminus N_{k_0}| \geq 2$ ) et  $B_{I'} = \cup_{j \leq k_0-1} A_{\sigma(j)} \cup B_{J_1}$  où  $B_{J_1}$  est obtenu à l'étape 11. Utilisons le lemme 1.1 (page 32) avec  $\mathcal{M}/(\cup_{j \leq k_0-1} A_{\sigma(j)})$ ,  $S = A_{\sigma(k_0)}$  et  $T = B_{J_1}$  pour obtenir  $OPT_{i_3}(\mathcal{M}/(\cup_{j \leq k_0-1} A_{\sigma(j)} \cup B_{J_1})) = OPT_{i_3}(\mathcal{M}/B_{I'}) \geq \frac{1}{n} OPT_{i_3}(\mathcal{M}) = \frac{|I'|}{n} OPT_{i_3}(\mathcal{M})$  car  $OPT_{i_3}(\mathcal{M}/(\cup_{j \leq k_0} A_{\sigma(j)})) < \frac{n-k_0}{n} OPT_{i_3}(\mathcal{M})$  et  $OPT_{i_3}(\mathcal{M}/(\cup_{j \leq k_0-1} A_{\sigma(j)})) \geq \frac{n-(k_0-1)}{n} OPT_{i_3}(\mathcal{M})$  puisque  $i_3 \in N_{k_0-1} \setminus N_{k_0}$  (correspondant à l'étape 18).

À l'étape 19, montrons que  $|N_{k_0-2} \setminus J_1| \leq 1$ . Par contradiction, supposons que  $|N_{k_0-2} \setminus J_1| > 1$ . Ainsi,

$$(n - k_0) + 2 + 1 < |J_1| + |N_{k_0-1} \setminus N_{k_0}| + |N_{k_0-2} \setminus (J_1 \cup N_{k_0-1})| \leq n - 2$$

car  $N_{k_0-2} \setminus (J_1 \cup N_{k_0-1}) = N_{k_0-2} \setminus J_1$  puisque par hypothèse nous sommes dans le cas où  $N_{k_0-2} \subseteq N_{k_0}$  et donc  $N_{k_0-2} \cap N_{k_0-1} \subseteq J_1$  par construction de  $J_1$ . D'où,  $k_0 > 5$ . Par ailleurs,

$$2 \leq |N_{k_0-1} \setminus N_{k_0}| \leq |N_{k_0-1}| \leq n - (k_0 - 1) - 3 \leq 9 - (k_0 - 1) - 3 = 7 - k_0$$

donc  $k_0 \leq 5$ , contradiction. Finalement  $|N_{k_0-2} \setminus J_1| \leq 1$ .

À l'étape 20, considérons le cas où  $J_4 = \{i_4\} = N_{k_0-2} \setminus J_1$ .

Par exemple, soient  $N = N_0 = \{1, \dots, 9\}$ ,  $N_1 = \{3, 4\}$ ,  $N_2 = \{4, 5, 6, 7\}$ ,  $N_3 = \{3, 8, 9\}$  et  $N_4 = \{3, 4, 5, 6, 7\}$ . Ici  $k_0 = 4$  et  $J_1 = \{2, 3, 4, 5, 6\}$ . Nous avons  $|N_{k_0-1} \setminus N_{k_0}| = |N_3 \setminus N_4| = 2$  et  $|N_{k_0-2} \setminus N_{k_0}| = |N_2 \setminus N_4| = 0$ , donc nous sommes aux étapes 17-21. On obtient  $J_3 = \{8\}$  et puisque  $N_{k_0-2} = N_2 \not\subseteq J_1$ , il s'ensuit que  $J_4 = \{i_4\} = N_2 \setminus J_1 = \{7\}$ .

il est évident que  $i_4 \notin N_{k_0-1}$  car sinon  $i_4$  serait dans  $J_1$  (par construction de  $J_1$ ). Posons  $I' = J_4$  et  $B_{I'} = \cup_{j \leq k_0-2} A_{\sigma(j)} \cup B_{J_1} \cup B_{J_3}$  où  $B_{J_1}$  est obtenu à l'étape 11 et  $B_{J_3}$  est obtenu à l'étape 18. Utilisons le lemme 1.1 (page 32) avec  $\mathcal{M}/(\cup_{j \leq k_0-2} A_{\sigma(j)})$ ,  $S = A_{\sigma(k_0-1)}$  et  $T = B_{J_1} \cup B_{J_3}$  pour obtenir  $OPT_{i_4}(\mathcal{M}/(\cup_{j \leq k_0-2} A_{\sigma(j)} \cup B_{J_1} \cup B_{J_3})) =$

$OPT_{i_4}(\mathcal{M}/B_{I'}) \geq \frac{1}{n}OPT_{i_4}(\mathcal{M}) = \frac{|I'|}{n}OPT_{i_4}(\mathcal{M})$  car  $OPT_{i_4}(\mathcal{M}/(\cup_{j \leq k_0-1} A_{\sigma(j)})) < \frac{n-(k_0-1)}{n}OPT_{i_4}(\mathcal{M})$  et  $OPT_{i_4}(\mathcal{M}/(\cup_{j \leq k_0-2} A_{\sigma(j)})) \geq \frac{n-(k_0-2)}{n}OPT_{i_4}(\mathcal{M})$  puisque  $i_4 \notin N_{k_0-1}$  et  $i_4 \in N_{k_0-2}$  (correspondant à l'étape 20).

À l'étape 21, nous savons que  $N \setminus (J_1 \cup J_3 \cup J_4) \subseteq N \setminus N_{k_0-2}$  ou de façon équivalente,  $N_{k_0-2} \subseteq J_1 \cup J_4$  car  $N_{k_0-2} \notin J_3$ ,  $|N_{k_0-2} \setminus N_{k_0}| = 0$ ,  $|N_{k_0-2} \setminus J_1| \leq 1$  et  $J_4 = N_{k_0-2} \setminus J_1$  lorsque  $|N_{k_0-2} \setminus J_1| = 1$ .

Soit  $I' = N \setminus (J_1 \cup J_3 \cup J_4)$  et  $B_{I'} = B_{J_1} \cup B_{J_3} \cup B_{J_4}$  où  $B_{J_1}$  est obtenu à l'étape 11,  $B_{J_3}$  est obtenu à l'étape 18 et  $B_{J_4}$  est obtenu à l'étape 20. Utilisons le lemme 1.1 (page 32) avec  $\mathcal{M}$ ,  $S = \cup_{j \leq k_0-2} A_{\sigma(j)}$  et  $T = B_{I'}$  afin d'obtenir  $\forall i \in I' \setminus \{i_1\}$ ,  $OPT_i(\mathcal{M}/B_{I'}) \geq \frac{k_0-2}{n}OPT_i(\mathcal{M}) = \frac{|I'|}{n}OPT_i(\mathcal{M})$  car  $OPT_i(\mathcal{M}/(\cup_{j \leq k_0-2} A_{\sigma(j)})) < \frac{n-(k_0-2)}{n}OPT_i(\mathcal{M})$  puisque  $I' \subseteq N \setminus N_{k_0-2}$  (correspondant à l'étape 21).

Aux étapes 22-31, forcément  $|N_{k_0-2} \setminus N_{k_0}| \geq 3$  car sinon nous serions aux étapes 17-21 (si  $|N_{k_0-2} \setminus N_{k_0}| = 0$ ) ou aux étapes 12-15 (si  $1 \leq |N_{k_0-2} \setminus N_{k_0}| \leq 2$ ). De plus,  $k_0 \geq 5$  (voir la preuve du lemme 5.1, page 187), donc  $N_{k_0-3}$  existe.

Si  $|N_{k_0-3} \setminus N_{k_0}| \leq 2$  alors nous sommes aux étapes 22-27 (l'exemple 5.2, page 132 décrit ce cas). Soit  $I' = J_5 = \{i_5, i'_5\} \subset N_{k_0-2} \setminus N_{k_0}$  (puisque  $|N_{k_0-2} \setminus N_{k_0}| \geq 3$ ) et  $B_{I'} = \cup_{j \leq k_0-2} A_{\sigma(j)} \cup B_{J_1}$  où  $B_{J_1}$  est obtenu à l'étape 11. Utilisons le lemme 1.1 (page 32) avec  $\mathcal{M}/(\cup_{j \leq k_0-2} A_{\sigma(j)})$ ,  $S = A_{\sigma(k_0-1)} \cup A_{\sigma(k_0)}$  et  $T = B_{J_1}$  afin d'obtenir  $\forall i \in I'$ ,  $OPT_i(\mathcal{M}/(\cup_{j \leq k_0-2} A_{\sigma(j)} \cup B_{J_1})) = OPT_i(\mathcal{M}/B_{I'}) \geq \frac{2}{n}OPT_i(\mathcal{M}) = \frac{|I'|}{n}OPT_i(\mathcal{M})$  car  $OPT_i(\mathcal{M}/(\cup_{j \leq k_0} A_{\sigma(j)})) < \frac{n-k_0}{n}OPT_i(\mathcal{M})$  et  $OPT_i(\mathcal{M}/(\cup_{j \leq k_0-2} A_{\sigma(j)})) \geq \frac{n-(k_0-2)}{n}OPT_i(\mathcal{M})$  puisque  $i \in N_{k_0-2} \setminus N_{k_0}$  (correspondant à l'étape 24).

À l'étape 25, montrons que  $|N_{k_0-3} \setminus (J_1 \cup J_5)| \leq 1$ . Par contradiction, supposons que  $|N_{k_0-3} \setminus (J_1 \cup J_5)| > 1$ . Ainsi,

$$(n - k_0) + 2 + 1 < |J_1| + |J_5| + |N_{k_0-3} \setminus (J_1 \cup J_5)| \leq n - 2$$

donc  $k_0 > 5$ . Par ailleurs,

$$3 \leq |N_{k_0-2} \setminus N_{k_0}| \leq |N_{k_0-2}| \leq n - (k_0 - 2) - 3 \leq 9 - (k_0 - 2) - 3 = 8 - k_0$$

donc  $k_0 \leq 5$ , contradiction. Finalement  $|N_{k_0-3} \setminus (J_1 \cup J_5)| \leq 1$ .

À l'étape 25, considérons le cas où  $J_6 = \{i_6\} = N_{k_0-3} \setminus (J_1 \cup J_5)$ . Il est évident que  $i_6 \notin N_{k_0-2}$  car sinon  $i_6$  serait dans  $J_5$  et par construction,  $J_5$  contient un nombre maximum d'éléments appartenant à  $N_{k_0-3} \setminus N_{k_0}$ . Posons  $I' = J_6$  et  $B_{I'} = \cup_{j \leq k_0-3} A_{\sigma(j)} \cup B_{J_1} \cup B_{J_5}$  où  $B_{J_1}$  est obtenu à l'étape 11 et  $B_{J_5}$  est obtenu à l'étape 24. Utilisons le lemme 1.1 (page 32) avec  $\mathcal{M}/(\cup_{j \leq k_0-3} A_{\sigma(j)})$ ,  $S = A_{\sigma(k_0-2)}$  et  $T = B_{J_1} \cup B_{J_5}$  pour obtenir  $OPT_{i_6}(\mathcal{M}/(\cup_{j \leq k_0-3} A_{\sigma(j)} \cup B_{J_1} \cup B_{J_5})) = OPT_{i_6}(\mathcal{M}/B_{I'}) \geq \frac{1}{n}OPT_{i_6}(\mathcal{M}) = \frac{|I'|}{n}OPT_{i_6}(\mathcal{M})$  car  $OPT_{i_6}(\mathcal{M}/(\cup_{j \leq k_0-2} A_{\sigma(j)})) < \frac{n-(k_0-2)}{n}OPT_{i_6}(\mathcal{M})$  et  $OPT_{i_6}(\mathcal{M}/(\cup_{j \leq k_0-3} A_{\sigma(j)})) \geq \frac{n-(k_0-3)}{n}OPT_{i_6}(\mathcal{M})$  puisque  $i_6 \notin N_{k_0-2}$  et  $i_6 \in N_{k_0-3}$  (correspondant à l'étape 26).



À l'étape 27, nous savons que  $N \setminus (J_1 \cup J_5 \cup J_6) \subseteq N \setminus N_{k_0-3}$  ou de façon équivalente  $N_{k_0-3} \subseteq (J_1 \cup J_5 \cup J_6)$  car  $|N_{k_0-3} \setminus (J_1 \cup J_5)| \leq 1$  et  $J_6 = N_{k_0-3} \setminus (J_1 \cup J_5)$  lorsque  $|N_{k_0-3} \setminus (J_1 \cup J_5)| = 1$ .

Soit  $I' = N \setminus (J_1 \cup J_5 \cup J_6)$  et  $B_{I'} = B_{J_1} \cup B_{J_5} \cup B_{J_6}$  où  $B_{J_1}$  est obtenu à l'étape 11,  $B_{J_5}$  est obtenu à l'étape 24 et  $B_{J_6}$  est obtenu à l'étape 26. Utilisons le lemme 1.1 (page 32) avec  $\mathcal{M}$ ,  $S = \cup_{j \leq k_0-3} A_{\sigma(j)}$  et  $T = B_{I'}$  pour obtenir  $\forall i \in I' \setminus \{i_1\}$ ,  $OPT_i(\mathcal{M}/B_{I'}) \geq \frac{k_0-3}{n} OPT_i(\mathcal{M}) = \frac{|I'|}{n} OPT_i(\mathcal{M})$  car  $OPT_i(\mathcal{M}/(\cup_{j \leq k_0-3} A_{\sigma(j)})) < \frac{n-(k_0-3)}{n} OPT_i(\mathcal{M})$  puisque  $I' \subseteq N \setminus N_{k_0-3}$  (correspondant à l'étape 27).

Aux étapes 28-31,  $|N_{k_0-3} \setminus N_{k_0}| \geq 3$  et puisque le cas  $|N_{k_0-3} \setminus N_{k_0}| \geq 4$  ne se produira jamais, on conclut que  $|N_{k_0-3} \setminus N_{k_0}| = 3$  (voir la preuve du lemme 5.1, page 187). Soit  $I' = J_7 = N_{k_0-3} \setminus N_{k_0}$  et  $B_{I'} = \cup_{j \leq k_0-3} A_{\sigma(j)} \cup B_{J_1}$  où  $B_{J_1}$  est obtenu à l'étape 11. Utilisons le lemme 1.1 (page 32) avec  $\mathcal{M}/(\cup_{j \leq k_0-3} A_{\sigma(j)})$ ,  $S = A_{\sigma(k_0-2)} \cup A_{\sigma(k_0-1)} \cup A_{\sigma(k_0)}$  et  $T = B_{J_1}$  pour obtenir  $\forall i \in I'$ ,  $OPT_i(\mathcal{M}/(\cup_{j \leq k_0-3} A_{\sigma(j)} \cup B_{J_1})) = OPT_i(\mathcal{M}/B_{I'}) \geq \frac{3}{n} OPT_i(\mathcal{M}) = \frac{|I'|}{n} OPT_i(\mathcal{M})$  car  $OPT_i(\mathcal{M}/(\cup_{j \leq k_0} A_{\sigma(j)})) < \frac{n-k_0}{n} OPT_i(\mathcal{M})$  et  $OPT_i(\mathcal{M}/(\cup_{j \leq k_0-3} A_{\sigma(j)})) \geq \frac{n-(k_0-3)}{n} OPT_i(\mathcal{M})$  puisque  $i \in N_{k_0-3} \setminus N_{k_0}$  (correspondant à l'étape 30).

À l'étape 31, montrons que  $N \setminus (J_1 \cup J_7) \subseteq N \setminus N_{k_0-3}$  ou de façon équivalente  $N_{k_0-3} \subset (J_1 \cup J_7)$ . Nous savons que  $J_7 = N_{k_0-3} \setminus N_{k_0}$ , donc il reste à montrer que  $(N_{k_0-3} \cap N_{k_0}) \subset J_1$ . Par construction de  $J_1$  (voir l'étape 10), ceci est équivalent à montrer que  $|J_1| = n - k_0 \geq |\{i_2\}| + |N_{k_0-1} \cap N_{k_0}| + |(N_{k_0-2} \cap N_{k_0}) \setminus N_{k_0-1}| + |(N_{k_0-3} \cap N_{k_0}) \setminus (N_{k_0-1} \cup N_{k_0-2})|$ . Par contradiction, supposons que

$$n - k_0 < |\{i_2\}| + |N_{k_0-1} \cap N_{k_0}| + |(N_{k_0-2} \cap N_{k_0}) \setminus N_{k_0-1}| + |(N_{k_0-3} \cap N_{k_0}) \setminus (N_{k_0-1} \cup N_{k_0-2})| \quad (D.4)$$

Nous savons que

- $|N_{k_0-1} \cap N_{k_0}| = |N_{k_0-1}| - |N_{k_0-1} \setminus N_{k_0}| \leq (n - (k_0 - 1) - 3) - 2 = n - k_0 - 4$  car  $|N_{k_0-1} \setminus N_{k_0}| \geq 2$ .
- $|(N_{k_0-2} \cap N_{k_0}) \setminus N_{k_0-1}| \leq |N_{k_0-2} \cap N_{k_0}| = |N_{k_0-2}| - |N_{k_0-2} \setminus N_{k_0}| \leq (n - (k_0 - 2) - 3) - 3 = n - k_0 - 4$  car  $|N_{k_0-2} \setminus N_{k_0}| \geq 3$ .
- $|(N_{k_0-3} \cap N_{k_0}) \setminus (N_{k_0-1} \cup N_{k_0-2})| \leq |N_{k_0-3} \cap N_{k_0}| = |N_{k_0-3}| - |N_{k_0-3} \setminus N_{k_0}| \leq (n - (k_0 - 3) - 3) - 3 = n - k_0 - 3$  car  $|N_{k_0-3} \setminus N_{k_0}| = 3$ .

En utilisant (D.4), on obtient  $n - k_0 < 1 + (n - k_0 - 4) + (n - k_0 - 4) + (n - k_0 - 3)$ , donc  $k_0 < n - 5 \leq 9 - 5 = 4$  qui est une contradiction avec  $k_0 - 3 \geq 1$  (puisque  $N_{k_0-3}$  existe), donc l'hypothèse (D.4) est fautive. Par construction de  $J_1$ ,  $N_{k_0-3} \subset J_1$  et  $N \setminus (J_1 \cup J_7) \subseteq N \setminus N_{k_0-3}$ .

Soit  $I' = N \setminus (J_1 \cup J_7)$  et  $B_{I'} = B_{J_1} \cup B_{J_7}$  où  $B_{J_1}$  est obtenu à l'étape 11 et  $B_{J_7}$  est obtenu à l'étape 30. Utilisons le lemme 1.1 (page 32) avec  $\mathcal{M}$ ,  $S = \cup_{j \leq k_0-3} A_{\sigma(j)}$  et  $T = B_{I'}$  pour obtenir  $\forall i \in I' \setminus \{i_1\}$ ,  $OPT_i(\mathcal{M}/B_{I'}) \geq \frac{k_0-3}{n} OPT_i(\mathcal{M}) = \frac{|I'|}{n} OPT_i(\mathcal{M})$  car  $OPT_i(\mathcal{M}/(\cup_{j \leq k_0-3} A_{\sigma(j)})) < \frac{n-(k_0-3)}{n} OPT_i(\mathcal{M})$  puisque  $I' \subseteq N \setminus N_{k_0-3}$  (correspondant à l'étape 31). ■

# Annexe **E**

## Abréviations et notations

### E.1 Abréviations

**EPTAS** : Efficient Polynomial Time Approximation Scheme

**FPTAS** : Fully Polynomial Time Approximation Scheme

**MA** : Multi-Agents

**MO** : Multi-Objectif

**OWA** : Ordered Weighted Averaging

**PTAS** : Polynomial Time Approximation Scheme

**SMA** : Système Multi-Agents

**TSP** : Traveling Salesman Problem

### E.2 Notations

$2^X$  : collection des parties de  $X$

$\mathcal{A}$  : algorithme

$\text{APX}$  : classe des problèmes d'optimisation admettant des algorithmes d'approximation à rapport constant

$\alpha$  : utilité maximum d'un élément

$\alpha_i$  : utilité maximum de l'agent  $i$  pour un élément

$B$  : base d'un matroïde

$\{B_1, \dots, B_n\}$  : base-allocation d'un matroïde

$\mathcal{B}$  : collection des bases d'un matroïde

$\mathcal{B}_u$  : collection des bases maximums pour  $u$

$C(\mathcal{M}, F)$  : complément de  $F$  dans le matroïde  $\mathcal{M}$  (voir page 32)

$C^{\max}(\mathcal{M}, F, u)$  : complément de  $F$  maximum pour  $u$  dans le matroïde  $\mathcal{M}$

$C^{\min}(\mathcal{M}, F, u)$  : complément de  $F$  minimum pour  $u$  dans le matroïde  $\mathcal{M}$

- $\delta$  : cocycle d'un sommet  
 $\delta^-$  : ensemble des arcs incidents intérieurement à un sommet  
 $\Delta$  : degré maximum des sommets d'un graphe  
 $e$  : élément d'un matroïde (ou bien indivisible)  
 $\epsilon$  : rapport d'approximation (ou la performance garantie)  
 $\epsilon_i$  : rapport d'approximation de l'objectif  $i$   
 $F$  : ensemble indépendant d'un matroïde  
 $\mathcal{F}$  : collection des ensembles indépendants d'un matroïde  
 $F + e$  : fait référence à  $F \cup \{e\}$   
 $F - e$  : fait référence à  $F \setminus \{e\}$   
 $g$  : fonction gain des couleurs  
 $G$  : graphe  
 $I$  : instance  
 $l_i$  : couleur (ou *label*)  $i$   
 $\mathcal{L}$  : fonction couleur (ou *label*)  
 $m$  : nombre d'éléments d'un matroïde (ou nombre de biens indivisibles)  
 $\mathcal{M}$  : matroïde  
 $\mathcal{M}_{\leq k}$  :  $k$ -troncature du matroïde  $\mathcal{M}$  (voir page 26)  
 $\mathcal{M}|X'$  : restriction du matroïde  $\mathcal{M}$  à  $X'$  (page 26)  
 $\mathcal{M}/X'$  : contraction du matroïde  $\mathcal{M}$  par  $X'$  (page 26)  
 $n$  : nombre d'agents  
 $[n]$  : fait référence à  $\{1, \dots, n\}$   
 $N$  : ensemble d'agents  
 $\mathbf{IN}$  : ensemble des entiers naturels  
 $\mathbb{NP}$  : classe des problèmes résolubles en temps polynomial sur une machine de Turing non déterministe  
 $obj$  : prend les valeur min ou max  
 $OPT$  : valeur optimale d'un problème d'optimisation  
 $OPT_i$  : valeur optimale de l'objectif  $i$  d'un problème d'optimisation multi-objectif  
 $OPT_i(\mathcal{M})$  : utilité maximale de l'agent  $i$  pour une base du matroïde  $\mathcal{M}$   
 $\mathbb{P}$  : classe des problèmes résolubles en temps polynomial sur une machine de Turing déterministe

- 
- $P_I$  : ensemble approché du front de Pareto de l'instance  $I$  d'un problème d'optimisation multi-objectif  
 $PAR(I)$  : ensemble des solutions Pareto-optimales de l'instance  $I$  d'un problème d'optimisation multi-objectif  
 $PIR(I)$  : valeur de la pire solution pour l'instance  $I$  d'un problème d'optimisation mono-objectif  
 $\pi$  : problème d'optimisation mono-objectif  
 $\pi_k$  : problème d'optimisation avec  $k$  objectifs  
 $r$  : fonction rang  
 $r(\mathcal{M})$  : rang du matroïde  $\mathcal{M}$   
 $R$  : ressource (ou ensemble de biens indivisibles)  
 $\mathcal{R}$  : collection des allocations admissibles  
 $\mathcal{R}_{\max}$  : collection des allocations admissibles maximales pour l'inclusion  
 $\mathbf{R}$  : ensemble des réels  
 $\mathbf{R}^+$  : ensemble des réels positifs  
 $S = \{S_1, \dots, S_n\}$  : allocation réalisable de biens indivisibles  
 $\Sigma_2^p$  : fait référence à la classe  $\mathbf{NP}^{\mathbf{NP}}$   
 $\theta$  : complexité de l'oracle d'indépendance d'un matroïde  
 $u$  : fonction d'utilité  
 $u_i$  : fonction d'utilité de l'agent  $i$   
 $V_n$  : fonction décroissante définie par morceaux sur  $[0, 1]$  et prend ses valeurs dans  $[0, \frac{1}{n}]$  pour un entier  $n \geq 2$  (voir la définition 2.3, page 63)  
 $VAL(I, x)$  : valeur de la solution  $x$  pour l'instance  $I$  d'un problème d'optimisation mono-objectif  
 $VAL_i(I, x)$  : valeur de la solution  $x$  sur l'objectif  $i$  pour l'instance  $I$  d'un problème d'optimisation multi-objectif  
 $w$  : fonction poids  
 $W_n$  : fonction décroissante définie par morceaux sur  $[0, 1]$  et prend ses valeurs dans  $[0, \frac{1}{n}]$  pour un entier  $n \geq 1$  (voir la définition 4.1, page 101)  
 $x$  : solution réalisable  
 $X$  : ensemble des éléments d'un matroïde  
 $X_I$  : ensemble des solutions réalisables de l'instance  $I$   
 $Z_{n,k}$  : fonction décroissante définie par morceaux sur  $[0, 1]$  et prend ses valeurs dans  $[0, \frac{1}{n}]$  pour des entiers  $n \geq 2$  et  $n > k \geq 1$  (voir page 66)  
 $\mathbb{Z}^+$  : ensemble des entiers relatifs positifs



## Liste des problèmes abordés

<b>Problème 1</b>	3-DIMENSIONAL MATCHING (3DM) (problème [SP1] dans Garey et Johnson (1979))
<i>Instance</i>	des ensembles $X, Y$ et $Z$ disjoints contenant le même nombre d'éléments $q$ et un ensemble $M \subseteq X \times Y \times Z$ .
<i>Question</i>	existe-t-il un ensemble $M' \subseteq M$ de taille $ M'  = q$ tel que pour tout couple de triplets $(x_1, y_1, z_1) \in M$ et $(x_2, y_2, z_2) \in M$ , $x_1 \neq x_2, y_1 \neq y_2$ et $z_1 \neq z_2$ ?
<b>Problème 2</b>	3-PARTITION (problème [SP15] dans Garey et Johnson (1979))
<i>Instance</i>	un ensemble $E$ de $3k$ éléments, une borne $b \in \mathbb{Z}^+$ et une fonction poids $w(e) \in \mathbb{Z}^+$ pour chaque $e \in E$ où $b/4 < w(e) < b/2$ et $\sum_{e \in E} w(e) = kb$ .
<i>Question</i>	l'ensemble $E$ peut-il être partitionné en $k$ ensembles $E_1, \dots, E_k$ tels que $ E_i  = 3$ et $\sum_{e \in E_i} w(e) = b$ pour tout $i \in [k]$ ?
<b>Problème 3</b>	ARBORESCENCE ENRACINÉE DE POIDS MAXIMUM (chapitre 6 de Korte et Vygen (2008))
<i>Instance</i>	un graphe $G = (V, E)$ orienté et connexe, une racine $r \in V$ et une fonction poids $w : E \rightarrow \mathbb{R}$ .
<i>Tâche</i>	une arborescence $\mathcal{T} \subseteq E$ (ensemble d'arcs constituant un arbre où chaque sommet a au plus un arc incident intérieurement) de poids $w(\mathcal{T}) = \sum_{e \in \mathcal{T}} w(e)$ maximum.
<b>Problème 4</b>	ARBRE COUVRANT AVEC UN NOMBRE MAXIMUM DE COULEURS (Broersma et Li, 1997)
<i>Instance</i>	un graphe $G = (V, E)$ non orienté et connexe et une fonction couleur $\mathcal{L} : E \rightarrow \{\ell_1, \dots, \ell_p\}$ .
<i>Tâche</i>	un arbre couvrant $T \subseteq E$ (graphe partiel connexe et sans cycle) avec $ \mathcal{L}(T)  =  \bigcup_{e \in T} \mathcal{L}(e) $ maximum.

Annexe F. Liste des problèmes abordés

<b>Problème 5</b>	ARBRE COUVRANT DE POIDS MAXIMUM (chapitre 6 de Korte et Vygen (2008))
<i>Instance</i>	un graphe $G = (V, E)$ non orienté et connexe et une fonction poids $w : E \rightarrow \mathbb{R}$ .
<i>Tâche</i>	un arbre couvrant $T$ de poids $w(T) = \sum_{e \in T} w(e)$ maximum.
<b>Problème 6</b>	BASE DE POIDS MAXIMUM (chapitre 13 de Korte et Vygen (2008))
<i>Instance</i>	un matroïde pondéré $\mathcal{M} = (X, \mathcal{F}, u)$ .
<i>Tâche</i>	une base $B \in \mathcal{F}$ de poids $u(B) = \sum_{e \in B} u(e)$ maximum.
<b>Problème 7</b>	BASE D'UN MATROÏDE MULTI-AGENTS (chapitres 2, 3, 4 et 5)
<i>Instance</i>	un matroïde $\mathcal{M} = (X, \mathcal{F})$ , un ensemble $N$ de $n \geq 1$ agents, $n$ fonctions d'utilité additives $u_i : X \rightarrow \mathbb{R}^+$ , $i = 1, \dots, n$ .
<i>Tâche</i>	une base-allocation $B = \{B_1, \dots, B_n\}$ qui satisfait un certain critère d'équité.
<b>Problème 8</b>	BASE D'UN MATROÏDE MULTI-OBJECTIF (chapitres 2, 3, 4 et 5)
<i>Instance</i>	un matroïde $\mathcal{M} = (X, \mathcal{F})$ , un entier $n \geq 1$ et des fonctions objectifs additives $u_i : X \rightarrow \mathbb{R}^+$ , $i = 1, \dots, n$ .
<i>Tâche</i>	une base $B \in \mathcal{F}$ qui maximise $u_i(B)$ , $i = 1, \dots, n$ .
<b>Problème 9</b>	BASE D'UN MATROÏDE PONDÉRÉ COLORIÉ (chapitre 6)
<i>Instance</i>	un matroïde $\mathcal{M} = (X, \mathcal{F})$ , une fonction poids $w : X \rightarrow \mathbb{R}^+$ , un entier $p \geq 1$ , une fonction couleur $\mathcal{L} : X \rightarrow \{\ell_1, \dots, \ell_p\}$ et une fonction gain des couleurs $g : \{\ell_1, \dots, \ell_p\} \rightarrow \mathbb{R}^+$ .
<i>Tâche</i>	une base $B \in \mathcal{F}$ qui maximise $w(B) = \sum_{e \in B} w(e)$ et $g(B) = \sum_{l \in \bigcup_{e \in B} \{\mathcal{L}(e)\}} g(l)$ .
<b>Problème 10</b>	PLUS COURT CHEMIN (chapitre 7 de Korte et Vygen (2008))
<i>Instance</i>	un graphe $G = (V, E)$ , une fonction longueur $\ell : E \rightarrow \mathbb{N}$ et deux sommets $a, b \in V$ .
<i>Tâche</i>	un chemin $P \subseteq E$ entre $a$ et $b$ de longueur $\ell(P) = \sum_{e \in P} \ell(e)$ minimum.
<b>Problème 11</b>	CHEMIN HAMILTONIEN DANS UN GRAPHE ORIENTÉ (problème [GT39] dans Garey et Johnson (1979))
<i>Instance</i>	un graphe orienté $G = (V, E)$ .

---

<i>Question</i>	le graphe $G$ contient-il un chemin hamiltonien (un chemin qui passe par chaque sommet exactement une fois) ?
-----------------	---

---

<b>Problème 12</b>	COLORATION DES ARÊTES D'UN GRAPHE (problème [GT4] dans Garey et Johnson (1979))
<i>Instance</i>	un graphe non orienté $G = (V, E)$ .
<i>Tâche</i>	colorier les arêtes de $G$ avec un nombre minimum de couleurs de sorte que deux arêtes adjacentes n'aient pas la même couleur.

---

<b>Problème 13</b>	COUPLAGE DE POIDS MAXIMUM (chapitre 11 de Korte et Vygen (2008))
<i>Instance</i>	un graphe $G = (V, E)$ non orienté et une fonction $w : E \rightarrow \mathbb{R}$ .
<i>Tâche</i>	un sous-ensemble d'arêtes $M \subseteq E$ disjointes de poids $w(M) = \sum_{e \in M} w(e)$ maximum.

---

<b>Problème 14</b>	GAP (Papadimitriou et Yannakakis, 2000)
<i>Instance</i>	un problème $\pi_k$ à $k \geq 2$ objectifs à minimiser, un vecteur de valeurs $v$ de dimension $k$ et une constante $\epsilon > 0$ .
<i>Tâche</i>	soit calculer une solution qui Pareto-domine $v$ , soit montrer qu'il n'existe pas de solution qui $(1 + \epsilon)$ -approche $v$ .

---

<b>Problème 15</b>	INTERSECTION DE DEUX MATROÏDES DE CARDINALITÉ MAXIMUM (chapitre 13 de Korte et Vygen (2008))
<i>Instance</i>	deux matroïdes $\mathcal{M}_1 = (X, \mathcal{F}_1)$ et $\mathcal{M}_2 = (X, \mathcal{F}_2)$ .
<i>Tâche</i>	un ensemble indépendant $F \in \mathcal{F}_1 \cap \mathcal{F}_2$ tel que $ F $ est maximum.

---

<b>Problème 16</b>	INTERSECTION DE DEUX MATROÏDES DE POIDS MAXIMUM (chapitre 13 de Korte et Vygen (2008))
<i>Instance</i>	deux matroïdes pondérés $\mathcal{M}_1 = (X, \mathcal{F}_1, u)$ et $\mathcal{M}_2 = (X, \mathcal{F}_2, u)$ .
<i>Tâche</i>	un ensemble indépendant $F \in \mathcal{F}_1 \cap \mathcal{F}_2$ de poids $w(F) = \sum_{e \in F} w(e)$ maximum.

---

<b>Problème 17</b>	INTERSECTION DE TROIS MATROÏDES DE CARDINALITÉ MAXIMUM (problème [SP11] dans Garey et Johnson (1979))
<i>Instance</i>	trois matroïdes $\mathcal{M}_1 = (X, \mathcal{F}_1)$ , $\mathcal{M}_2 = (X, \mathcal{F}_2)$ et $\mathcal{M}_3 = (X, \mathcal{F}_3)$ .
<i>Tâche</i>	un ensemble indépendant $F \in \mathcal{F}_1 \cap \mathcal{F}_2 \cap \mathcal{F}_3$ tel que $ F $ est maximum.

---

<b>Problème 18</b>	$k$ -COUVERTURE DE POIDS MAXIMUM (Nemhauser <i>et al.</i> , 1978)
<i>Instance</i>	un entier $k \geq 1$ , une collection $\{S_1, \dots, S_m\}$ d'ensembles et un poids $w(e)$ associé à chaque élément $e \in \bigcup_{j=1}^m S_j$ .



Annexe F. Liste des problèmes abordés

---

<i>Tâche</i>	un sous-ensemble $J \subseteq \{1, \dots, m\}$ de taille $ J  \leq k$ tel que le poids des éléments couverts $\sum_{e \in (\bigcup_{j \in J} S_j)} w(e)$ est maximum.
<b>Problème 19</b>	PARTAGE DE GÂTEAU (CAKE-CUTTING) (Brams et Taylor, 1996)
<i>Instance</i>	Un gâteau représenté par l'intervalle $[0, 1]$ , un ensemble $N$ de $n \geq 1$ agents, chaque agent $i \in N$ a une fonction d'utilité $u_i$ associant à chaque $I \subseteq [0, 1]$ une utilité $u_i(I) \geq 0$ .
<i>Tâche</i>	Partitionner l'intervalle $[0, 1]$ en $n$ parties $I_1, \dots, I_n$ où $I_i$ revient à l'agent $i \in \{1, \dots, n\}$ en satisfaisant un certain critère d'équité.
<b>Problème 20</b>	PARTITION (problème [SP12] dans Garey et Johnson (1979))
<i>Instance</i>	un ensemble d'éléments $A = \{a_1, \dots, a_m\}$ et une fonction poids $s(a) \in \mathbb{Z}^+, \forall a \in A$ .
<i>Question</i>	existe-t-il un ensemble $A' \subseteq A$ tel que $\sum_{a \in A'} s(a) = \sum_{e \in A \setminus A'} s(a)$ ?
<b>Problème 21</b>	PROBLÈME CLASSIQUE D'ALLOCATION DE BIENS INDIVISIBLES (Brams et Taylor, 1996)
<i>Instance</i>	un ensemble $R$ de $m \geq 0$ biens indivisibles, un ensemble $N$ de $n \geq 1$ agents et des utilités $u_i : 2^R \rightarrow \mathbb{R}^+, i = 1, \dots, n$ .
<i>Tâche</i>	une allocation $S = \{S_1, \dots, S_n\}$ (partition de $R$ ) qui satisfait un certain critère d'équité.
<b>Problème 22</b>	PROBLÈME MA (chapitre 2)
<i>Instance</i>	un ensemble $R$ , une collection $\mathcal{R} \subseteq 2^R$ , un ensemble $N$ de $n \geq 1$ agents et des utilités $u_i : 2^R \rightarrow \mathbb{R}^+, i = 1, \dots, n$ .
<i>Tâche</i>	un vecteur de $n$ ensembles deux à deux disjoints $(S_1, \dots, S_n)$ où $S = \bigcup_{i \in N} S_i \in \mathcal{R}_{\max} \subseteq \mathcal{R}$ en satisfaisant un critère d'équité.
<b>Problème 23</b>	PROBLÈME MO (chapitre 2)
<i>Instance</i>	un ensemble $R$ , une collection $\mathcal{R} \subseteq 2^R$ , un entier $n \geq 1$ , des fonctions objectifs $u_i : \mathcal{R}_{\max} \rightarrow \mathbb{R}^+, i = 1, \dots, n$ .
<i>Tâche</i>	trouver un ensemble $S \in \mathcal{R}_{\max}$ qui maximise $u_i(S), i = 1, \dots, n$ .
<b>Problème 24</b>	VOYAGEUR DE COMMERCE (TSP ou problème [ND22] dans Garey et Johnson (1979))
<i>Instance</i>	un ensemble $C$ de $m \geq 2$ villes et une distance $d(i, j) \in \mathbb{Z}^+$ pour chaque paire de villes $i, j \in C$ .
<i>Tâche</i>	un tour $T$ (cycle hamiltonien) de $C$ de distance totale $d(T) = \sum_{i \in T} d(i)$ minimum.

## Liste des illustrations

### G.1 Algorithmes et protocoles

1.1	GLOUTON (Edmonds, 1971) . . . . .	32
2.1	ALLOCATE (Markakis et Psomas, 2011) . . . . .	65
3.1	PROCHE DE L'ABSENCE DE JALOUSIE . . . . .	82
3.2	PROCHE DE LA PROPORTIONNALITÉ . . . . .	85
3.3	PROCHE DE LA PROPORTIONNALITÉ POUR LES AGENTS DE $N_1$ . . . . .	85
3.4	PROCHE DE L'ABSENCE D'ENVIE POUR DEUX AGENTS . . . . .	93
4.1	SEUIL . . . . .	104
4.2	MAX-SEUIL . . . . .	113
5.1	DIVISER-ET-CHOISIR . . . . .	126
5.2	DIVISER-INTERROGER-ET-CHOISIR . . . . .	128
5.3	PROTOCOLE . . . . .	130
5.4	DIVISER . . . . .	134
6.1	ALT-GLOUTON . . . . .	147
6.2	EXACT . . . . .	164

### G.2 Figures

1.1	Solutions $\epsilon$ -approchées pour un problème de maximisation. . . . .	12
1.2	Approche budget pour un problème de maximisation bi-objectif. . . . .	17
1.3	Approximation d'une solution en multi-objectif. . . . .	19
1.4	Approximation du point idéal pour un problème de maximisation bi-objectif. . . . .	21
2.1	$V_n$ pour $n = 2$ et $n = 3$ . . . . .	64

2.2	Les fonctions $\frac{k}{n}V_k\left(\frac{n}{k}x\right)$ et $V_n(x)$ pour $n \in \{2, 3\}$ et $k \in \{1, \dots, n-1\}$ .	66
3.1	Exemple 3.4. . . . .	92
4.1	$V_n$ et $W_n$ pour $n = 2$ et $n = 3$ . . . . .	102
4.2	Exemple 4.2 : le graphe $G$ . . . . .	105
4.3	Exemple 4.2 : le graphe contracté $G/B_1$ . . . . .	105
4.4	Les fonctions $W_2(\alpha)$ et $\overline{W}_2(\alpha)$ . . . . .	119
5.1	Arborescence du Protocole 5.3. . . . .	132
5.2	Exemple d'exécution du Protocole 5.3. . . . .	133
5.3	Les fonctions $\frac{k}{n}V_k\left(\frac{n}{k}x\right)$ et $W_n(x)$ pour $n \in \{2, 3\}$ et $k \in \{1, \dots, n-1\}$ .	139
6.1	Instance critique de ALT-GLOUTON pour des matroïdes coloriés-simples. . . . .	149
6.2	Instance critique de ALT-GLOUTON pour des matroïdes simples. . . . .	151
6.3	Instance critique de ALT-GLOUTON GÉNÉRALISÉ pour des matroïdes simples. . . . .	154
6.4	Le multigraphe $G_L$ . . . . .	160
6.5	Le multigraphe $G_2$ . . . . .	161
6.6	Exemple 6.7. . . . .	163

### G.3 Tableaux

2.1	Exemple 2.11. . . . .	58
3.1	Exemple 3.1. . . . .	83
4.1	Exemple 4.1. . . . .	99
4.2	Exemple 4.3. . . . .	106
4.3	Bornes supérieure et inférieure sur l'utilité de l'agent le plus défavorisé.	121
5.1	Exemple 5.1. . . . .	127

# Bibliographie

- AGGARWAL, V., ANEJA, Y. P. et NAIR, K. (1982). Minimal spanning tree subject to a side constraint. *Computers & Operations Research*, 9(4):287–296.
- ANDERSSON, M. R. et SANDHOLM, T. W. (1998). Contract types for satisficing task allocation : Ii experimental results. *In Proceedings of the AAAI spring symposium series : Satisficing models*, pages 1–7.
- ANGEL, E., BAMPIS, E. et GOURVÈS, L. (2005). Approximation results for a bicriteria job scheduling problem on a single machine without preemption. *Information processing letters*, 94(1):19–27.
- ANGEL, E., BAMPIS, E. et GOURVÈS, L. (2007). *Approximation in Multiobjective Problems*. Handbook of Approximation Algorithms and Metaheuristics, T. F. Gonzalez. Chapman & Hall/CRC.
- ASADPOUR, A., FEIGE, U. et SABERI, A. (2008). Santa claus meets hypergraph matchings. *In Approximation, Randomization and Combinatorial Optimization. Algorithms and Techniques*, pages 10–20. Springer.
- ASADPOUR, A. et SABERI, A. (2010). An approximation algorithm for max-min fair allocation of indivisible goods. *SIAM Journal on Computing*, 39(7):2970–2989.
- AUSIELLO, G. (1999). *Complexity and approximation : Combinatorial optimization problems and their approximability properties*. Springer.
- AZIZ, H., GASPERS, S., MACKENZIE, S. et WALSH, T. (2014). Fair assignment of indivisible objects under ordinal preferences. *In Proceedings of the 13th International Conference on Autonomous Agents and MultiAgent Systems (AAMAS'14)*, pages 1305–1312.
- BANSAL, N. et SVIRIDENKO, M. (2006). The santa claus problem. *In Proceedings of the 38th annual ACM Symposium on Theory Of Computing (STOC'06)*, pages 31–40.

- BAUMEISTER, D., BOUVERET, S., LANG, J., NGUYEN, N.-T., NGUYEN, T. T. et ROTHE, J. (2014). Scoring rules for the allocation of indivisible goods. *In Proceedings of the 21st European Conference on Artificial Intelligence (ECAI'14)*, pages 75–80.
- BAZGAN, C., GOURVÈS, L. et MONNOT, J. (2013). Approximation with a fixed number of solutions of some multiobjective maximization problems. *Journal of Discrete Algorithms*, 22:19–29.
- BERGER, A., BONIFACI, V., GRANDONI, F. et SCHÄFER, G. (2011). Budgeted matching and budgeted matroid intersection via the gasoline puzzle. *Mathematical Programming*, 128(1-2):355–372.
- BEZÁKOVÁ, I. et DANI, V. (2005). Allocating indivisible goods. *SIGecom Exchanges*, 5(3):11–18.
- BIXBY, R. E. et CUNNINGHAM, W. H. (1995). *Handbook of Combinatorics, chapter Matroid Optimization and Algorithms*. North Holland.
- BOUVERET, S. (2007). *Allocation et partage équitables de ressources indivisibles : modélisation, complexité et algorithmique*. Thèse de doctorat, Université de Toulouse, Institut Supérieur de l’Aéronautique et de l’Espace.
- BOUVERET, S., ENDRISS, U. et LANG, J. (2010). Fair division under ordinal preferences : Computing envy-free allocations of indivisible goods. *In Proceedings of the 19th European Conference on Artificial Intelligence (ECAI'10)*, pages 387–392.
- BOUVERET, S. et LANG, J. (2008). Efficiency and envy-freeness in fair division of indivisible goods : Logical representation and complexity. *Journal of Artificial Intelligence Research*, 32(1):525–564.
- BOUVERET, S. et LANG, J. (2011). A general elicitation-free protocol for allocating indivisible goods. *In Proceedings of the 22nd International Joint Conference on Artificial Intelligence (IJCAI'11)*, volume 22, page 73.
- BOUVERET, S. et LANG, J. (2014). Manipulating picking sequences. *In Proceedings of the 21st European Conference on Artificial Intelligence (ECAI'14)*, pages 141–146.
- BOUVERET, S., LEMAÎTRE, M., FARGIER, H. et LANG, J. (2005). Allocation of indivisible goods : a general model and some complexity results. *In Proceedings of the 4th International Conference on Autonomous Agents and MultiAgent Systems (AAMAS'05)*, pages 1309–1310. ACM.

- BOUVERET, S. et LEMAÎTRE, M. (2014). Characterizing conflicts in fair division of indivisible goods using a scale of criteria. *In Proceedings of the 13th International Conference on Autonomous Agents and MultiAgent Systems (AAMAS'14)*.
- BRAESS, P. D. D. D. (1968). Über ein paradoxon aus der verkehrsplanung. *Unternehmensforschung*, 12(1):258–268.
- BRAMS, S. J., EDELMAN, P. H. et FISHBURN, P. C. (2003). Fair division of indivisible items. *Theory and Decision*, 55(2):147–180.
- BRAMS, S. J. et TAYLOR, A. D. (1995). An envy-free cake division protocol. *The American Mathematical Monthly*, 102(1):9–18.
- BRAMS, S. J. et TAYLOR, A. D. (1996). *Fair division - from cake-cutting to dispute resolution*. Cambridge University Press.
- BROERSMA, H. et LI, X. (1997). Spanning trees with many or few colors in edge-colored graphs. *Discussiones Mathematicae Graph Theory*, 17(2):259–269.
- BROERSMA, H., LI, X., WOEGINGER, G. et ZHANG, S. (2005). Paths and cycles in colored graphs. *Australasian Journal of Combinatorics*, 31:299–311.
- BRUALDI, R. (1969). Comments on bases in dependence structures. *Bulletin of the Australian Mathematical Society*, 1:161–167.
- BRÜGGEMANN, T., MONNOT, J. et WOEGINGER, G. J. (2003). Local search for the minimum label spanning tree problem with bounded color classes. *Operations Research Letters*, 31(3):195–201.
- BRYLAWSKI, T. H. (1973). Some properties of basic families of subsets. *Discrete Mathematics*, 6(4):333 – 341.
- BUDISH, E. (2011). The combinatorial assignment problem : Approximate competitive equilibrium from equal incomes. *Journal of Political Economy*, 119(6):1061–1103.
- CAMERINI, P. M., GALBIATI, G. et MAFFIOLI, F. (1984). The complexity of multi-constrained spanning tree problems. *Theory of algorithms*, pages 53–101.
- CAMERINI, P. M. et MAFFIOLI, F. (1975). Bounds for 3-matroid intersection problems. *Information Processing Letters*, 3(3):81–83.
- CAMERINI, P. M., MAFFIOLI, F. et VERCELLIS, C. (1989). Multi-constrained matroidal knapsack problems. *Mathematical Programming*, 45(1-3):211–231.

- CAMERINI, P. M. et VERCELLIS, C. (1984). The matroidal knapsack : a class of (often) well-solvable problems. *Operations Research Letters*, 3(3):157–162.
- CARAGIANNIS, I., KAKLAMANIS, C., KANELLOPOULOS, P. et KYROPOULOU, M. (2012). The efficiency of fair division. *Theory of Computing Systems*, 50(4):589–610.
- CHAKRABARTY, D., CHUZHOU, J. et KHANNA, S. (2009). On allocating goods to maximize fairness. In *Proceedings of the 50th Annual IEEE Symposium on Foundations Of Computer Science (FOCS'09)*, pages 107–116.
- CHEKURI, C., VONDRAK, J. et ZENKLUSEN, R. (2010). Dependent randomized rounding via exchange properties of combinatorial structures. In *Proceedings of the 51st Annual IEEE Symposium on Foundations Of Computer Science (FOCS'10)*, pages 575–584.
- CHEVALEYRE, Y., DUNNE, P. E., ENDRISS, U., LANG, J., LEMAÎTRE, M., MAUDET, N., PADGET, J., PHELPS, S., RODRIGUEZ-AGUILAR, J. A. et SOUSA, P. (2006). Issues in multiagent resource allocation. *Informatica (Slovenia)*, 30(1).
- CHEVALEYRE, Y., ENDRISS, U., ESTIVIE, S. et MAUDET, N. (2007a). Reaching envy-free states in distributed negotiation settings. In *Proceedings of the 20th International Joint Conference on Artificial Intelligence (IJCAI'07)*, pages 1239–1244.
- CHEVALEYRE, Y., ENDRISS, U., ESTIVIE, S. et MAUDET, N. (2008). Multiagent resource allocation in k-additive domains : Preference representation and complexity. *Annals of Operations Research*, 163(1):49–62.
- CHEVALEYRE, Y., ENDRISS, U., LANG, J. et MAUDET, N. (2007b). A short introduction to computational social choice. In *Proceedings of the 33rd Conference on Current Trends in Theory and Practice of Computer Science (SOFSEM'07)*, pages 51–69.
- CHEVALEYRE, Y., ENDRISS, U. et MAUDET, N. (2007c). Allocating goods on a graph to eliminate envy. In *Proceedings of the 22nd AAAI Conference on Artificial Intelligence (AAAI'07)*, pages 700–705.
- COFFMAN, E. G. et LANGSTON, M. A. (1984). A performance guarantee for the greedy set-partitioning algorithm. *Acta Informatica*, 21(4):409–415.
- COOK, S. A. (1971). The complexity of theorem-proving procedures. In *Proceedings of the 3rd annual ACM Symposium on Theory Of Computing (STOC'71)*, pages 151–158.

- 
- CORMEN, T. H., LEISERSON, C. E., RIVEST, R. L., STEIN, C. *et al.* (2001). *Introduction to algorithms*, volume 2. MIT Press Cambridge.
- COUËTOUX, B., GOURVÈS, L., MONNOT, J. et TELELIS, O. (2010). Labeled traveling salesman problems : Complexity and approximation. *Discrete Optimization*, 7(1-2):74–85.
- CSIRIK, J., KELLERER, H. et WOEGINGER, G. (1992). The exact lpt-bound for maximizing the minimum completion time. *Operations Research Letters*, 11(5): 281–287.
- DE KEIJZER, B., BOUVERET, S., KLOS, T. et ZHANG, Y. (2009). On the complexity of efficiency and envy-freeness in fair division of indivisible goods with additive preferences. *In Proceedings of the 1st International Conference on Algorithmic Decision Theory (ADT'09)*.
- DEMKO, S. et HILL, T. P. (1988). Equitable distribution of indivisible objects. *Mathematical Social Sciences*, 16(2):145–158.
- DEURMEYER, B. L., FRIESEN, D. K. et LANGSTON, M. A. (1982). Scheduling to maximize the minimum processor finish time in a multiprocessor system. *SIAM Journal on Algebraic Discrete Methods*, 3(2):190–196.
- DOBZINSKI, S., NISAN, N. et SCHAPIRA, M. (2005). Approximation algorithms for combinatorial auctions with complement-free bidders. *In Proceedings of the 37th annual ACM Symposium on Theory Of Computing (STOC'05)*, pages 610–618.
- DUBINS, L. E. et SPANIER, E. H. (1961). How to cut a cake fairly. *American mathematical monthly*, 68:1–17.
- EDMONDS, J. (1971). Matroids and the greedy algorithm. *Mathematical programming*, 1(1):127–136.
- EDMONDS, J. (1979). Matroid intersection. *Annals of discrete Mathematics*, 4:39–49.
- EDMONDS, J. et FULKERSON, D. R. (1965). Transversals and matroid partition. *Journal of Research of the National Bureau of Standards - B. Mathematical Sciences*, 69:147–153.
- EHRGOTT, E. (1996). On matroids with multiple objectives. *Optimization*, 38(1): 73–84.
- EHRGOTT, M. (2005). *Multicriteria Optimization*. Springer.



- ENDRISS, U. et MAUDET, N. (2004). Welfare engineering in multiagent systems. *In the 4th International Workshop Engineering Societies in the Agents World IV (ESAW'03)*, pages 93–106. Springer.
- ENDRISS, U., MAUDET, N., SADRI, F. et TONI, F. (2006). Negotiating socially optimal allocations of resources. *Journal of Artificial Intelligence Research (JAIR)*, 25:315–348.
- ESCOFFIER, B., GOURVÈS, L. et MONNOT, J. (2013). Fair solutions for some multiagent optimization problems. *Journal of Autonomous Agents and Multi-Agent Systems*, 26(2):184–201.
- ESTIVIE, S. (2006). *Allocation de ressources multi-agents : théorie et pratique*. Thèse de doctorat, Université Paris-Dauphine.
- ESTIVIE, S., CHEVALEYRE, Y., ENDRISS, U. et MAUDET, N. (2005). Sur le caractère égalitaire de l'allocation de ressources distribuée. *In Actes des Journées Francophones sur les Systèmes Multi-Agents (JFSMA '05)*, pages 143–156.
- FEIGE, U. (1998). A threshold of  $\ln n$  for approximating set cover. *Journal of the ACM (JACM)*, 45(4):634–652.
- FERRAIOLI, D., GOURVÈS, L. et MONNOT, J. (2014). On regular and approximately fair allocations of indivisible goods. *In Proceedings of the 13th International Conference on Autonomous Agents and MultiAgent Systems (AAMAS'14)*, pages 997–1004.
- FISK, C. (1979). More paradoxes in the equilibrium assignment problem. *Transportation Research Part B : Methodological*, 13(4):305–309.
- FRANK, A. (1981). A weighted matroid intersection algorithm. *Journal of Algorithms*, 2(4):328–336.
- GABOW, H. N. et TARJAN, R. E. (1984). Efficient algorithms for a family of matroid intersection problems. *Journal of Algorithms*, 5(1):80–131.
- GALAND, L. (2008). *Méthodes exactes pour l'optimisation multicritère dans les graphes : recherche de solutions de compromis*. Thèse de doctorat, Université Paris VI.
- GAREY, M. R., GRAHAM, R. L. et ULLMAN, J. D. (1972). An analysis of some packing algorithms. *In Combinatorial Algorithms*, numéro 9, pages 39–47.
- GAREY, M. R. et JOHNSON, D. S. (1979). *Computers and intractability*, volume 174. W. H. Freeman and Company.

- GLASSER, C., REITWIESSNER, C., SCHMITZ, H. et WITEK, M. (2010). Hardness and approximability in multi-objective optimization. *In Electronic Colloquium on Computational Complexity (ECCC'10)*, volume 17, page 31.
- GOEL, A. et MEYERSON, A. (2006). Simultaneous optimization via approximate majorization for concave profits or convex costs. *Algorithmica*, 44(4):301–323.
- GOEMANS, M. X., HARVEY, N. J. A., IWATA, S. et MIRROKNI, V. (2009). Approximating submodular functions everywhere. *In Proceedings of the 20th Annual ACM-SIAM Symposium On Discrete Algorithms (SODA'09)*, pages 535–544.
- GOLFARELLI, M., MAIO, D. et RIZZI, S. (1997). A task-swap negotiation protocol based on the contract net paradigm. Rapport technique DEIS, CSITE, 005–97, Università di Bologna.
- GOLOVIN, D. (2005). Max-min fair allocation of indivisible goods. Rapport technique 2348, Computer Science Department, Carnegie Mellon University.
- GONDRAN, M. et MINOUX, M. (1995). *Graphes et algorithmes*. Eyrolles.
- GORSKI, J. (2010). *Multiple Objective Optimization and Implications for Single Objective Optimization*. Thèse de doctorat, Bergische Universität Wuppertal.
- GOURVÈS, L. (2005). *Approximation polynomiale et optimisation combinatoire multicritère*. Thèse de doctorat, Laboratoire de méthodes Informatiques de l'Université d'Evry Val d'Essone.
- GOURVÈS, L., MONNOT, J. et TLILANE, L. (2012). Approximate tradeoffs on matroids. *In Proceedings of the 20th European Conference on Artificial Intelligence (ECAI'12)*, pages 360–365.
- GOURVÈS, L., MONNOT, J. et TLILANE, L. (2013a). A matroid approach to the worst case allocation of indivisible goods. *In Proceedings of the 23rd International Joint Conference on Artificial Intelligence (IJCAI'13)*, pages 136–142.
- GOURVÈS, L., MONNOT, J. et TLILANE, L. (2013b). A protocol for cutting matroids like cakes. *In Proceedings of the 9th International Conference on Web and Internet Economics (WINE'13)*, pages 216–229.
- GOURVÈS, L., MONNOT, J. et TLILANE, L. (2014). Near fairness in matroids. *In Proceedings of the 21st European Conference on Artificial Intelligence (ECAI'14)*, pages 393–398.
- GRABISCH, M. (1997). k-order additive discrete fuzzy measures and their representation. *Fuzzy sets and systems*, 92(2):167–189.

- GRAHAM, R. L. (1966). Bounds for certain multiprocessing anomalies. *Bell System Technical Journal*, 45(9):1563–1581.
- GRAHAM, R. L. (1969). Bounds on multiprocessing timing anomalies. *SIAM Journal on Applied Mathematics*, 17(2):416–429.
- GRANDONI, F., RAVI, R. et SINGH, M. (2009). Iterative rounding for multi-objective optimization problems. In *Proceedings of the 17th Annual European Symposium on Algorithms (ESA'09)*, pages 95–106. Springer.
- GRANDONI, F. et ZENKLUSEN, R. (2010). Approximation schemes for multi-budgeted independence systems. In *Proceedings of the 18th Annual European Symposium on Algorithms (ESA'10)*, pages 536–548. Springer.
- GREENE, C. (1973). A multiple exchange property for bases. In *Proceedings of the American Mathematical Society*, pages 45–50.
- GREENE, C. et MAGNANTI, T. L. (Nov. 1975). Some abstract pivot algorithms. *SIAM Journal on Applied Mathematics*, 29(3):530–539.
- HAMACHER, H. W. et RUHE, G. (1994). On spanning tree problems with multiple objectives. *Annals of Operations Research*, 52(4):209–230.
- HANSEN, P. (1980). Bicriterion path problems. In *Multiple criteria decision making theory and application*, pages 109–127. Springer.
- HASSIN, R. (1992). Approximation schemes for the restricted shortest path problem. *Mathematics of Operations Research*, 17(1):36–42.
- HASSIN, R. et LEVIN, A. (2004). An efficient polynomial time approximation scheme for the constrained minimum spanning tree problem using matroid intersection. *SIAM Journal on Computing*, 33(2):261–268.
- HASSIN, R., MONNOT, J. et SEGEV, D. (2007). Approximation algorithms and hardness results for labeled connectivity problems. *Journal of Combinatorial Optimization*, 14(4):437–453.
- HERREINER, D. et PUPPE, C. (2002). A simple procedure for finding equitable allocations of indivisible goods. *Social Choice and Welfare*, 19(2):415–430.
- HILL, T. P. (1987). Partitioning general probability measures. *The Annals of Probability*, 15(2):804–813.
- HOCHBAUM, D. S. (1997). Approximation algorithms for NP-hard problems. *ACM SIGACT News*, 28(2):40–52.

- HONG, S.-P., CHUNG, S.-J. et PARK, B. H. (2004). A fully polynomial bicriteria approximation scheme for the constrained spanning tree problem. *Operations Research Letters*, 32(3):233–239.
- JAMAIN, F. (2014). *Représentations discrètes de l'ensemble des points non dominés pour des problèmes d'optimisation multi-objectifs*. Thèse de doctorat, Université Paris-Dauphine.
- KALINOWSKI, T., NARODYTSKA, N. et WALSH, T. (2013a). A social welfare optimal sequential allocation procedure. In *Proceedings of the 23rd international joint conference on Artificial Intelligence (IJCAI'13)*, pages 227–233.
- KALINOWSKI, T., NARODYTSKA, N., WALSH, T. et XIA, L. (2013b). Strategic behavior when allocating indivisible goods sequentially. In *Proceedings of the 27th AAAI Conference on Artificial Intelligence (AAAI'13)*.
- KHOT, S. et PONNUSWAMI, A. K. (2007). Approximation algorithms for the max-min allocation problem. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques*, pages 204–217. Springer.
- KORTE, B. et VYGEN, J. (2008). *Combinatorial Optimization : Theory and Algorithms*. Springer.
- KRUMKE, S. O. et WIRTH, H. C. (1998). On the minimum label spanning tree problem. *Information Processing Letters*, 66(2):81–85.
- LEE, J. et RYAN, J. (1992). Matroid applications and algorithms. *ORSA Journal on Computing*, 4(1):70–98.
- LEHMANN, B., LEHMANN, D. et NISAN, N. (2001). Combinatorial auctions with decreasing marginal utilities. In *Proceedings of the 3rd ACM conference on Electronic Commerce*, pages 18–28.
- LIPTON, R. J., MARKAKIS, E., MOSSEL, E. et SABERI, A. (2004). On approximately fair allocations of indivisible goods. In *Proceedings of the 5th ACM Conference on Electronic Commerce (EC'04)*, pages 125–131.
- LORENZ, D. H. et RAZ, D. (2001). A simple efficient approximation scheme for the restricted shortest path problem. *Operations Research Letters*, 28(5):213–219.
- LU, T. et BOUTILIER, C. (2011). Budgeted social choice : From consensus to personalized decision making. In *Proceedings of the 22nd International Joint Conference on Artificial Intelligence (IJCAI'11)*, pages 280–286.

- LUMET, C. (2012). *Modélisation, représentation et résolution de problèmes de partage équitable de biens indivisibles soumis au risque*. Thèse de doctorat, Institut Supérieur de l’Aéronautique et de l’Espace (ISAE).
- MAFFIOLI, F., RIZZI, R. et BENATI, S. (2007). Least and most colored bases. *Discrete Applied Mathematics*, 155(15):1958–1970.
- MARATHE, M. V., RAVI, R., SUNDARAM, R., RAVI, S. S., ROSENKRANTZ, D. J. et HUNT III, H. B. (1998). Bicriteria network design problems. *Journal of algorithms*, 28(1):142–171.
- MARKAKIS, E. et PSOMAS, C.-A. (2011). On worst-case allocations in the presence of indivisible goods. In *Proceedings of the 7th Workshop on Internet and Network Economics (WINE’11)*, pages 278–289. Springer.
- MONNOT, J., PASCHOS, V. T. et TOULOUSE, S. (2003). *Approximation polynomiale des problèmes NP-difficiles : optima locaux et rapport différentiel*. Hermès science publications.
- MOULIN, H. (2004). *Fair division and collective welfare*. MIT Press.
- NASH, J. F. (1950). The bargaining problem. *Econometrica : Journal of the Econometric Society*, pages 155–162.
- NEMHAUSER, G. L., WOLSEY, L. A. et FISHER, M. L. (1978). An analysis of approximations for maximizing submodular set functions-I. *Mathematical Programming*, 14(1):265–294.
- NGUYEN, T. T. (2013). *Approximability and Inapproximability of Social Welfare Optimization in Multiagent Resource Allocation*. Thèse de doctorat, der Heinrich-Heine-Universität Düsseldorf.
- NGUYEN, T. T., ROOS, M. et ROTHE, J. (2013). A survey of approximability and inapproximability results for social welfare optimization in multiagent resource allocation. *Annals of Mathematics and Artificial Intelligence*, 68(1-3):65–90.
- NGUYEN, T. T. et ROTHE, J. (2013a). Envy-ratio and average-nash social welfare optimization in multiagent resource allocation. In *Proceedings of the 12th International Conference on Autonomous Agents and MultiAgent Systems (AAMAS)*, pages 1139–1140.
- NGUYEN, T. T. et ROTHE, J. (2013b). How to decrease the degree of envy in allocations of indivisible goods. In *Proceedings of the 3rd International Conference on Algorithmic Decision Theory (ADT’13)*, pages 271–284.

- ORPONEN, P. et MANNILA, H. (1987). On approximation preserving reductions : complete problems and robust measures. Rapport technique C-1987-28, Department of Computer Science, University of Helsinki.
- OXLEY, J. (1992). *Matroid Theory*. Oxford University Press.
- PAPADIMITRIOU, C. H. et YANNAKAKIS, M. (1991). Optimization, approximation, and complexity classes. *Journal of computer and system sciences*, 43(3):425–440.
- PAPADIMITRIOU, C. H. et YANNAKAKIS, M. (2000). *On the Approximability of Trade-offs and Optimal Access of Web Sources*. Proceedings of the 41st Annual Symposium on Foundations Of Computer Science (FOCS'00).
- PARKES, D. C., PROCACCIA, A. D. et SHAH, N. (2012). Beyond dominant resource fairness : extensions, limitations, and indivisibilities. *In Proceedings of the 13th ACM Conference on Electronic Commerce (EC'12)*, pages 808–825.
- PASCHOS, V. (2004). *Complexité et approximation polynomiale*. Lavoisier.
- PERNY, P., SPANJAARD, O. et STORME, L.-X. (2006). A decision-theoretic approach to robust optimization in multivalued graphs. *Annals of Operations Research*, 147(1):317–341.
- PIGOU, A. C. (1924). *The economics of welfare*. MacMillan, London.
- PROCACCIA, A. D. (2013). Cake cutting : not just child's play. *Communications of the ACM*, 56(7):78–87.
- PROCACCIA, A. D., ROSENSCHEIN, J. S. et ZOHAR, A. (2008). On the complexity of achieving proportional representation. *Social Choice and Welfare*, 30(3):353–362.
- PROCACCIA, A. D. et WANG, J. (2014). Fair enough : Guaranteeing approximate maximin shares. *In Proceedings of the 15th Conference on Economics and Computation (EC'14)*.
- RAVI, R. et GOEMANS, M. X. (1996). The constrained minimum spanning tree problem. *In Proceedings of the 5th Scandinavian Workshop on Algorithmic Theory (SWAT'96)*, pages 66–75. Springer.
- RAVI, R., MARATHE, M. V., RAVI, S. S., ROSENKRANTZ, D. J. et HUNT III, H. B. (1993). Many birds with one stone : Multi-objective approximation algorithms. *In Proceedings of the 25th annual ACM Symposium on Theory Of Computing (STOC'93)*, pages 438–447.

- ROBERTSON, J. M. et WEBB, W. A. (1998). *Cake-cutting algorithms - be fair if you can*. A K Peters Ltd.
- SAFER, H. et ORLIN, J. (1995). *Fast Approximation Schemes for Multi-Criteria Combinatorial Optimization*. Sloan School of Management, Massachusetts Institute of Technology.
- SAHNI, S. et GONZALEZ, T. (1976). P-complete approximation problems. *Journal of the ACM (JACM)*, 23(3):555–565.
- SANDHOLM, T. (1993). An implementation of the contract net protocol based on marginal cost calculations. In *Proceedings of the 7th AAAI Conference on Artificial Intelligence (AAAI'93)*, pages 256–262.
- SANDHOLM, T. (1998). Contract types for satisficing task allocation : I theoretical results. In *Proceedings of the AAAI spring symposium series : Satisficing models*, pages 23–25.
- SCHRIJVER, A. (2003). *Combinatorial Optimization : Polyhedra and Efficiency*. Springer.
- SERAFINI, P. (1987). Some considerations about computational complexity for multi objective combinatorial problems. In *Recent advances and historical development of vector optimization*, pages 222–232. Springer.
- SHORROCKS, A. F. (1983). Ranking income distributions. *Economica*, pages 3–17.
- SIMON, H. U. (1990). On approximate solutions for combinatorial optimization problems. *SIAM Journal on Discrete Mathematics*, 3(2):294–310.
- SKOWRON, P., FALISZEWSKI, P. et SLINKO, A. (2013). Fully proportional representation as resource allocation : Approximability results. In *Proceedings of the 23rd International Joint Conference on Artificial Intelligence (IJCAI'13)*, pages 353–359.
- SMITH, R. G. (1980). The contract net protocol : High-level communication and control in a distributed problem solver. *IEEE Transactions on Computers*, C-29(12):1104–1113.
- STEIN, C. et WEIN, J. (1997). On the existence of schedules that are near-optimal for both makespan and total weighted completion time. *Operations Research Letters*, 21(3):115–122.
- STEINHAUS, H. (1948). The problem of fair division. *Econometrica*, 16:101–104.

- SUZUMURA, K., ARROW, K. J. et SEN, A. K. (2010). *Handbook of Social Choice & Welfare*, volume 2. Elsevier.
- TANG, L. et ZHANG, P. (2012). Approximating minimum label s-t cut via linear programming. In *Proceedings of the 10th Latin American international conference on Theoretical INformatics (LATIN'12)*, pages 655–666.
- VASSILVITSKII, S. et YANNAKAKIS, M. (2005). Efficiently computing succinct trade-off curves. *Theoretical Computer Science*, 348(2):334–356.
- VAZIRANI, V. V. (2001). *Approximation algorithms*. Springer.
- VIZING, V. G. (1964). On an estimate of the chromatic class of a p-graph. *Diskret. Analiz*, 3(7):25–30.
- VONDRÁK, J. (2008). Optimal approximation for the submodular welfare problem in the value oracle model. In *Proceedings of the 40th annual ACM Symposium on Theory Of Computing (STOC'08)*, pages 67–74.
- WARBURTON, A. (1985). Worst case analysis of greedy and related heuristics for some min-max combinatorial optimization problems. *Mathematical Programming*, 33(2):234–241.
- WARBURTON, A. (1987). Approximation of pareto optima in multiple-objective, shortest-path problems. *Operations Research*, 35(1):70–79.
- WEISS, G. (2013). *Multiagent systems : a modern approach to distributed artificial intelligence*. MIT Press.
- WEYMARK, J. A. (1981). Generalized gini inequality indices. *Mathematical Social Sciences*, 1(4):409–430.
- WHITNEY, H. (1935). On the abstract properties of linear dependence. *American Journal of Mathematics*, 57(3):509–533.
- WILLIAMSON, D. P. et SHMOYS, D. B. (2011). *The design of approximation algorithms*. Cambridge University Press.
- WOEGINGER, G. J. (1997). A polynomial-time approximation scheme for maximizing the minimum machine completion time. *Operations Research Letters*, 20(4):149–154.
- WOODALL, D. R. (1974). An exchange theorem for bases of matroids. *Journal of Combinatorial Theory (B)*, 16:227 – 228.



## Bibliographie

---

- WOOLDRIDGE, M. (2009). *An introduction to multiagent systems*. John Wiley & Sons.
- YOUNG, H. P. (1995). *Equity : in theory and practice*. Princeton University Press.
- ZHANG, P., CAI, J. Y., TANG, L. et ZHAO, W. (2011). Approximation and hardness results for label cut and related problems. *Journal of Combinatorial Optimization*, 21(2):192–208.

## **Les matroïdes et leur implication dans l'allocation de ressources indivisibles : algorithmes d'approximation avec garantie de performance**

**Résumé :** Nous nous intéressons dans cette thèse à la problématique de la décision collective. L'objectif est de déterminer une solution de compromis pour des problèmes soumis à de multiples points de vue. Les problèmes considérés sont de nature combinatoire. Plus précisément, il s'agit de la classe des systèmes d'ensembles qui ont une structure de matroïde. La théorie des matroïdes est centrale en optimisation combinatoire, elle a permis d'unifier des structures apparemment séparées comme les arbres et les couplages dans les graphes et elle a engendré des algorithmes efficaces pour résoudre des problèmes d'optimisation non triviaux en temps polynomial. Nous nous intéressons à fournir des algorithmes d'approximation polynomiaux centralisés et décentralisés avec garantie de performance pour déterminer une solution de compromis qui est une base du matroïde. La solution de compromis doit également être équitable pour tous les membres de la collectivité. Nous portons un intérêt particulier au problème de partage équitable de biens indivisibles qui est une thématique importante en choix social computationnel et dont le problème se modélise par les matroïdes.

**Mots clés :** Optimisation combinatoire, approximation polynomiale à garantie de performance, matroïdes, allocation de biens indivisibles, notions d'équité.

## **Matroids and their implication in the allocation of indivisible resources : approximation algorithms with guaranteed performance**

**Abstract :** In this thesis, we are interested in collective decision-making. The objective is to find a tradeoff solution for problems that are evaluated by multiple points of view. We consider problems having a matroid structure. Matroid theory is significant in combinatorial optimization, it helped to unify apparently separated structures like forests and matchings in graphs and it includes efficient algorithms for solving non trivial optimization problems in polynomial time. We are interested to provide polynomial time centralized and decentralized approximation algorithms for finding a tradeoff solution which is a base of the matroid. The tradeoff solution must also be fair for all the members of the community. We are particularly interested in the issue of the fair division of indivisible goods which is central in computational social choice and that can be modeled by matroids.

**Key words :** Combinatorial optimization, polynomial time approximation with guaranteed performance, matroids, allocation of indivisible goods, fairness notions.