



HAL
open science

Étude des notions de diversité et d'envie dans le cadre de problèmes d'affectation avec groupes d'agents

Nathanaël Gross-Humbert

► **To cite this version:**

Nathanaël Gross-Humbert. Étude des notions de diversité et d'envie dans le cadre de problèmes d'affectation avec groupes d'agents. Intelligence artificielle [cs.AI]. Sorbonne Université, 2024. Français. NNT : 2024SORUS257 . tel-04810038

HAL Id: tel-04810038

<https://theses.hal.science/tel-04810038v1>

Submitted on 28 Nov 2024

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 n° 130 : Informatique, Télécommunications et Electronique
Sorbonne Université, Paris

Étude des notions de diversité et d'envie dans le cadre de problèmes d'affectation avec groupes d'agents

Thèse de doctorat en informatique
présentée publiquement par

Nathanaël GROSS–HUMBERT

le 4 septembre 2024

Rapporteurs

GRANDI Umberto

Professeur, Université Toulouse Capitole

PINO PÉREZ Ramón

Professeur, Université d'Artois

Examineurs

LANG Jérôme

Directeur de recherche, Université Paris Dauphine - PSL

WILCZYNSKI Anaëlle

Maîtresse de conférences, CentraleSupélec

Directrice de thèse

BEYNIER Aurélie

Maîtresse de conférences

Co-encadrants de thèse

BENABBOU Nawal

Maîtresse de conférences

MAUDET Nicolas

Professeur

Sorbonne Université, France

Président du jury: LANG Jérôme

Table of contents

Remerciements	5
Introduction	7
1 État de l'art	11
1.1 Problèmes d'affectation et variantes	12
1.1.1 Problème d'appariement	13
1.1.2 Ressources divisibles et indivisibles	18
1.1.3 Types d'agents	22
1.2 Qualité des solutions	23
1.2.1 Notion d'efficacité	23
1.2.2 Notions d'équité	27
1.2.3 Mécanismes optimisant l'équité	30
1.2.4 Mécanismes optimisant efficacité et équité	31
1.3 Diversité	32
1.3.1 Conclusion	35
2 Stabilité sous contraintes de diversité	37
2.1 Introduction	37
2.2 Modèle	38
2.2.1 Instances et Allocations	38
2.2.2 Utilité des agents, utilité globale	41
2.2.3 Amplitude maximum de l'utilité	43
2.3 Prix de la stabilité	46
2.4 Mécanisme de loterie	50
2.5 Procédure d'échanges bilatéraux	54
2.6 Expérimentations	59
2.6.1 Prix de la stabilité empirique	59
2.6.2 Comparaison des résultats sur les données réelles de l'immobilier de Singapour	63
2.6.3 Nombre d'échanges avant terminaison	64
2.7 Conclusion du chapitre	66

3	Envie entre types d'agents	69
3.1	Introduction	69
3.2	Allocations de ressources entre groupes d'agents	70
3.3	Axiomatisation des notions d'envie	73
3.3.1	Définitions des axiomes	73
3.3.2	Notions d'envie existantes	75
3.3.3	Vérifications des axiomes	79
3.4	Une nouvelle famille de notions d'envie	83
3.4.1	Envie pour au moins un sous-groupe	84
3.4.2	Envie pour tout sous-groupe	89
3.4.3	Envie pour k sous-groupes	96
3.5	Retour sur la notion de monotonie	102
3.5.1	Monotonie de l'envie	103
3.5.2	Monotonie de population et de ressources	110
3.6	Conclusion du chapitre	112
4	Minimisation du degré d'envie entre types d'agents	115
4.1	Introduction	115
4.2	Modèle	116
4.3	Degré d'envie	117
4.3.1	Définition	117
4.3.2	Complexité de calcul	118
4.4	Approximation du degré d'envie	120
4.4.1	Réduction à une variante du problème de sac à dos	120
4.4.2	Approximation par un FPRAS	122
4.4.3	Méthode d'échantillonnage	123
4.5	Expériences	138
4.6	Conclusion du chapitre	140
	Conclusion	143
	Bibliographie	148
A	Construction des permutations pour la preuve de la proposition 2.5.2	161
A.1	Cas k impair	161
A.2	Cas k pair	162
B	Preuve de la proposition 2.3.2	165
B.1	Preuve que $1.6 \leq PoS$	165
B.2	Preuve que $PoS \leq 2$	166

Remerciements

Je souhaite en premier lieu remercier mes encadrants, Nawal Benabbou, Aurélie Beynier, et Nicolas Maudet, pour m'avoir accompagné tout au long de cette thèse et du stage de M2 qui a précédé. Grâce à leurs efforts, je me suis senti accueilli et soutenu pendant les 4 années que j'ai passées dans leurs équipes. Je remercie Aurélie Beynier tout d'abord, pour avoir été ma directrice de thèse, pour m'avoir tutoré pendant mes travaux et m'avoir communiqué sa rigueur et son savoir-faire. Je remercie ensuite Nawal Benabbou, pour sa patience et son aide dans les rédactions de mes preuves parfois difficile à comprendre. Je remercie enfin Nicolas Maudet, pour ses conseils et ses suggestions, toujours pertinentes, et pour m'avoir fait profiter de sa perspicacité dans mes travaux.

Je remercie Umberto Grandi et Ramón Pino Pérez de bien avoir voulu être les rapporteurs de ma thèse, ainsi que Jérôme Lang et Anaëlle Wilczynski, pour avoir été mes examinateurs. Je remercie également Jérôme Lang, encore, et Fanny Pascual pour avoir constitué mon comité de suivi.

Je remercie le service informatique du LIP6, et plus particulièrement Vincent Cuzin, pour son assistance lors de l'utilisation du cluster de calcul.

Je remercie les doctorants du couloir 25-26 4ème étage, avec qui j'ai passé toutes ces années de thèses. Ceux de l'équipe Terraneon, Maël Franceschetti, Tristan Bersoux et Laura Echeverri, collègues de bureaux et de celui d'à côté, avec qui je discutais chaque midi. Jordan Thieyre, Fares Chouaki et Manuel Amoussou, qui m'ont accueilli dans leur bureau quand j'ai dû quitter le mien. Je remercie également les multiples doctorants en CIFRE, qui ont constitué des rencontres irrégulières mais des plus intéressantes.

Je remercie mes amis de l'école des ponts et chaussées, que j'ai vu soutenir avant moi, pour leur soutien. Je remercie également ma famille, et tout particulièrement mon père, pour ses conseils, et pour m'avoir aidé à m'orienter dans cette voie.



Introduction

La question du partage équitable se retrouve dans des situations aussi nombreuses que variées, du partage d'un gâteau entre plusieurs personnes (Brams et Taylor [1996]) à la répartition des corvées (Aziz et Rey [2020]), de l'attribution des places en écoles aux candidats (Aziz et Sun [2021]) à la division du loyer entre colocataires (Airiau *et al.* [2023]). Même si ces problèmes comportent d'importantes différences en termes de contexte ou de ressources considérées, ils présentent des problématiques très similaires : il s'agit dans tous les cas de trouver une manière de partager une certaine quantité de ressources (dénombrables ou non) entre plusieurs personnes d'une manière que l'on considérera "équitable". Il n'est alors pas étonnant de voir que si le partage équitable en tant que domaine d'étude scientifique ne commence à se démarquer qu'au début des années 50 (avec Steinhaus [1948]), il existe des résultats bien plus anciens, et il est possible de retrouver des méthodes de partage équitable jusque dans l'ancien testament. Le domaine du partage équitable couvre l'étude des différentes manières de représenter la notion d'équité au moyen de propriétés théoriques, telles que la proportionnalité (Steinhaus [1948]) et l'envie (Foley [1967]), ainsi que les divers algorithmes permettant de garantir ces propriétés. Puisque les problèmes considérés sont variés, les modèles utilisés pour les représenter le sont également, ce qui engendre une grande diversité de notions et de méthodes, chacune pertinente pour un ou plusieurs modèles particuliers. Ces notions sont étudiées suivant un point de vue global, où l'on cherche à faire en sorte que l'état du système respecte certaines propriétés (comme la proportionnalité ou le minmax fair share), soit suivant un point de vue local, où l'on considère les comportements individuels des agents (pour vérifier si par exemple une méthode est résistante aux manipulations).

Un domaine lié à celui du partage équitable est celui du choix social computationnel, né au début des années 2000 des progrès des machines de calcul, qui regroupe les considérations algorithmiques et computationnelles relatives aux problèmes classiques de décisions collectives (un tour d'horizon du domaine a été effectué par Bouveret *et al.* [2016]). Si le partage équitable comportait un aspect algorithmique depuis ses débuts en tant que domaine d'étude scientifique (avec par exemple les travaux de Gale et Shapley [1962]), l'émergence du choix social computationnel accrût l'importance des algorithmes dans la littérature. De plus, une nouvelle considération apparut : la question de la complexité computationnelle de ces algorithmes. En effet, le temps de calcul nécessaire à l'exécution d'une méthode peut nuancer des résultats théoriques positifs

comme négatifs : une durée d'exécution déraisonnable peut invalider la pertinence d'un algorithme, tandis qu'au contraire une méthode peut être considérée comme résistante à la manipulation si les seules méthodes de manipulation possibles sont trop longues pour être applicables en pratique.

Les travaux présentés dans ce manuscrit reposent sur un modèle inspiré du problème de l'immobilier de Singapour (décrit par Benabbou *et al.* [2018]). Dans ce problème, un organisme public en charge de la répartition des logements sociaux souhaite allouer un certain nombre de logements à des demandeurs. Le problème de l'immobilier de Singapour n'est pas unique et il existe de nombreux problèmes structurellement semblables. Cependant, nous disposons pour ce cas précis d'un jeu de données réelles, ce qui nous permet d'effectuer des expériences sur des instances proches des circonstances réelles. De toute évidence, notre modèle représentera les demandeurs par des agents et les logements par des ressources à partager. Il sera également nécessaire de considérer certaines contraintes et dynamiques pour représenter les spécificités de la situation. Ces contraintes sont similaires à celles qui se retrouvent dans la plupart des problèmes de *house allocation* (comme Zhou [1990] et Abdulkadiroğlu et Sönmez [1998]). Pour commencer, il est impossible de découper un logement entre plusieurs familles, ces ressources sont donc indivisibles. De plus, un seul logement doit être alloué à chaque demandeur, ce qui réduit la marge de manœuvre : dans certains problèmes, un lot de plusieurs objets de basse qualité peuvent être équivalents à un unique objet de bonne qualité, mais ce n'est pas le cas dans notre problème, où plusieurs logements ne peuvent pas être combinés en un "meilleur" logement. Les logements ne sont, bien sûr, pas identiques et peuvent varier en termes de taille, location, prix du loyer, et de tout autre critère de qualité considéré par les agents. Certains de ces critères sont subjectifs (une même location peut-être appropriée ou non selon les besoins de l'agent), tandis que même ceux en apparence objectifs, tels que le prix du loyer, peuvent avoir une importance plus ou moins grande pour un agent. Par exemple, un agent riche sera moins réticent à payer un loyer élevé, tandis que ce dernier peut être rendre un logement inenvisageable pour un agent plus pauvre. Par conséquent, nous ne chercherons pas à représenter dans le modèle les caractéristiques des logements, et nous conférons plutôt à chaque agent une fonction d'utilité, qui à chaque logement associe une mesure globale de la qualité du logement aux yeux de l'agent en question. Les caractéristiques des logements et les priorités des agents sont alors "cachées" derrière ces fonctions : par exemple, un agent souhaitant un appartement de grande taille attribuera simplement une valeur d'utilité plus importante à ces appartements. Les préférences seront bien sûr à sens unique (dite "one sided") : les logements n'auront pas d'opinion sur leurs occupants. En cela, notre modèle diffère de problèmes tels que le choix des écoles (étudié par exemple par Echenique et Yenmez [2015]), où les élèves se voient attribuer des places en écoles en fonction de leurs préférences et de celles des écoles (les préférences sont alors dite "two-sided"). Nous souhaitons de plus représenter le fait que nos agents ne forment pas un groupe homogène, mais puissent provenir d'origines sociales ou ethniques diverses. Nous organiserons donc nos agents par *type* pouvant représenter diverses caractéristiques de l'agent (ethnie, classe sociale, etc...). Par souci de simplicité,

nous supposerons que chaque agent appartient à un unique type. Nous nous servirons de ces types pour étudier les questions d'équité et de diversité.

En plus de la qualité de l'appartement, nous souhaiterons parfois ajouter aux agents des préférences sur leurs voisinages : les agents préfèrent habiter dans un quartier où leurs voisins leur ressemblent. Cette notion d'homophilie est inspirée de celle utilisée dans les jeux de Schelling (introduit par Schelling [1969]) et dans certains jeux de coalitions hédoniques (Bredereck *et al.* [2019]), un modèle multi-agent cherchant à modéliser l'évolution de la mixité des populations dans les villes. Nous organiserons alors nos ressources de manière similaire à nos agents, en utilisant des *blocs* représentant la proximité géographique (maisons d'un quartier, appartements d'un immeuble, etc...). Comme pour les groupes, nous supposerons que chaque ressource appartient à un et un seul type par souci de simplicité. La composition du voisinage, c'est-à-dire quels agents sont alloués aux logements du même bloc, sera alors prise en compte dans les préférences de l'agent, en plus de la qualité du logement lui-même.

Une fois les objectifs des agents explicités, nous souhaitons étudier ce qui se passe lorsque les agents sont autorisés à modifier localement l'état du système. En effet, si la génération d'allocations est souvent considérée de manière globale, c'est-à-dire imposée, ou du moins dirigée, par une autorité centrale, en pratique la longévité d'une telle allocation peut dépendre de l'intérêt qu'ont les agents à la maintenir. Nous allons donc permettre aux agents d'améliorer leur situation en échangeant leurs logements entre eux par paire, sous réserve bien sûr que les deux y aient intérêt. Au niveau de la littérature, cette dynamique peut se rapprocher de certains modèles de jeux de Schelling (par exemple ceux utilisés par Chauhan *et al.* [2018]), elle peut également être liée à certaines notions d'équité, notamment la notion d'envie. Au niveau des situations réelles, cette dynamique a non seulement été implémentée dans certains cas (site d'échange de parcelles agricoles ¹ ou d'immobilier en Île de France ²), mais elle est parfois spontanément apparue lorsque des agents ont voulu échanger leurs allocations (deux lycéens néerlandais ont voulu échanger leur place en école, événement couvert par Ronald de Haan [2017]). De plus, des dynamiques semblables ont déjà été étudié dans la littérature (Endriss *et al.* [2006]), y compris dans le cadre de problèmes de *house allocation* (Damamme *et al.* [2015]). Nous considérerons alors la notion de stabilité : une allocation sera stable si elle est "résistante" à cette dynamique, c'est-à-dire si les agents ne souhaitent pas ou ne peuvent pas modifier l'allocation. De toute évidence, une allocation stable est préférable à une allocation que les agents auront la volonté et la capacité de modifier.

Si l'équité est l'un des objectifs principaux de toute étude relevant du partage équitable, il existe de nombreuses définitions possibles de ce qu'une allocation doit satisfaire pour être considérée "équitable". Puisque nous considérons des comportements individuels entre agents, il serait approprié de choisir une définition qui ne requiert pas un point de vue global pour être appréhendée. La notion d'envie, introduite par Foley [1967], est alors particulièrement pertinente, puisqu'elle se concentre sur le point de vue de

¹<https://www.echangeparcelle.fr>

²<https://www.echangerhabiter.fr>

l'agent. Son concept est simple : un agent sera envieux si il ne préfère pas l'allocation d'un autre agent à sa propre allocation. Bien sûr, cet agent n'a aucun moyen de savoir ce que ses comparses pensent de leurs propres allocations, il devra donc utiliser ses propres préférences pour juger. La question de trouver des allocations ne contenant pas d'agent envieux a été l'objet de nombreux travaux (par exemple Caragiannis *et al.* [2019], Budish [2011], Halpern *et al.* [2020], ou encore Bu *et al.* [2023]). À noter que comme il n'est pas toujours possible de trouver une allocation sans envie (prouvé par Caragiannis *et al.* [2019]), on se tournera parfois vers une relaxation (telle que celles introduite par Budish [2011] et Caragiannis *et al.* [2019]). Une spécificité de notre cas est que nos agents sont répartis en différents types, ce qui soulève naturellement la question : comment vérifier si une allocation est équitable entre les différents types ? Le premier réflexe serait de simplement considérer comment les types estiment les allocations combinées des agents d'un autre type, mais ce genre de définition perd en pertinence lorsque les types peuvent regrouper des nombres d'agents différents : imaginons un type comprenant deux agents et un type comprenant cinq agents, sur quelles bases peuvent-ils se comparer? Cette question devient d'autant plus complexe que notre modèle n'autorise qu'une seule ressource allouée à chaque agent : une seule famille par logement, un seul logement par famille. Non seulement cela force les lots de ces types à être déséquilibrés l'un par rapport à l'autre, mais cela pose aussi la question de comment deux agents doivent juger cinq logements, et vice-versa. De nombreux travaux se sont penchés sur ce problème (comme Benabbou *et al.* [2018], Chakraborty *et al.* [2021], Scarlett *et al.* [2023] et Aleksandrov et Walsh [2018a]), proposant diverses solutions que nous comparerons.

Le plan de ce manuscrit suivra le fil du raisonnement que nous venons de discuter. Après un tour d'horizon de la littérature du domaine dans le chapitre 1, nous commencerons par étudier dans le chapitre 2 une extension du modèle de l'immobilier de Singapour, qui se démarque de l'original en ajoutant aux objectifs des agents un critère d'homophilie. Nous considérerons de plus une approche nouvelle du modèle, puisque nous allons étudier une dynamique décentralisée exécutée par les agents et la comparer aux méthodes de résolution classiques. Nous allons ensuite, dans le chapitre 3, aborder la question de l'équité en utilisant la notion d'envie. Nous allons discuter de la pertinence des définitions existantes dans le cadre de notre modèle, puis en proposer de nouvelles. Un aspect que nous avons mentionné mais dont nous avons peu discuté est l'aspect computationnel. Le chapitre 4 sera donc dédié à l'étude des algorithmes permettant de calculer les mesures introduites dans le chapitre 3 et de leur complexité, afin de pouvoir conclure sur la faisabilité pratique des méthodes présentées. Enfin, le dernier chapitre sera dédié à la conclusion, où nous ferons le points sur les résultats présentés et discuterons des perspectives qui s'en dégagent.

Chapitre 1

État de l'art

Les problèmes d'allocations constituent une famille de problèmes consistant à générer et étudier des allocation sur des ensembles d'agent. De fait, cette description est extrêmement générale, par conséquent, de nombreux problèmes provenant de domaines très divers peuvent être placés dans cette catégorie. On y retrouvera ainsi des problèmes d'économie (tels que divers problèmes de marchés), d'informatique (tels que l'ordonnancement des tâches), ou du domaine qui nous intéresse, le choix social computationnel et plus particulièrement du partage équitable (Bouveret *et al.* [2016]). Puisqu'ils sont destinés à résoudre une grande variété de problèmes, les modèles utilisés peuvent présenter un grand nombre de différences. Un problème d'allocation peut allouer des ressources à des agents ou des agents entre eux, les ressources peuvent être divisibles ou indivisibles, les agents peuvent avoir des préférences ordinales ou cardinales, les ressources peuvent avoir des préférences sur les agents ou non, les agents peuvent être répartis en types, ces considérations sont les principales différences que l'on retrouve dans la littérature. Bien d'autres sont possibles, mais nous allons ici présenter des exemples des principaux archétypes de problèmes d'allocations.

Outre les différences de modèle, les différents travaux sur les problèmes d'allocations se démarquent par leur objectifs, c'est-à-dire quels sont les qualités qui sont attendues des allocations. Le plus simple de ces objectifs est l'efficacité : on souhaite que les agents se voient alloués les objets qu'ils préfèrent. Il existe bien sûr plusieurs critères possibles pour qu'une allocation soit considérée comme "efficace". Un second objectif est l'équité : on souhaite qu'aucun agent ne soit lésé. Cet objectif est plus complexe à conceptualiser, et plusieurs définitions basées sur plusieurs idées de ce que l'on considère "équitable" ont vu le jour. Divers mécanismes et algorithmes ont alors été conçus pour essayer de satisfaire ces deux objectifs, parfois de manière simultanée.

La notion de diversité se démarque en étant à mi-chemin entre un objectif et un choix de modélisation. En effet, si sa présence résulte d'une volonté en amont que les allocations satisfassent un critère de diversité, cette notion sera en pratique souvent intégrée au modèle en tant que contrainte. Nous verrons dans la suite les différentes manières dont la diversité a été implémentée.

1.1 Problèmes d'affectation et variantes

La première grande différence entre les modèles sera la forme que vont prendre les préférences, celle-ci pouvant être ordinaire (un ordre sur les lot possible) ou cardinale (une valeur attribuée à chaque lot possible). Contrairement aux différences que nous verront dans la suite, qui sont souvent liées les unes aux autres, la forme des préférences est quasiment indépendante de tout autre choix de modélisation et la plupart des problèmes peuvent utiliser indifféremment des utilités ordinaires ou cardinales. Si nous allons présenter dans le cadre de cet état de l'art des exemples de modèles avec les deux formes de préférences, nous ne considérerons dans la suite que les préférences cardinales, c'est-à-dire que chaque agent i disposera d'une fonction d'utilité u_i qui à chaque allocation possible lui renvoie une valeur réelle. Nous nous limiterons de plus dans le cadre de ce manuscrit à des valeurs d'utilité positive ou nulles. Cette hypothèse est courante dans la littérature, mais certains problèmes particuliers autorisent des utilités négatives (comme par exemple les problèmes de répartitions des biens et corvées (Aziz *et al.* [2019a]; Aziz et Rey [2020]). En outre, pour le cas des préférences cardinales, la très grande majorité de la littérature se restreint à des fonctions d'utilités additives, c'est-à-dire que pour tous lots d'objets B_1 et B_2 , les fonctions d'utilités u seront telles que $u(B_1 \cup B_2) = u(B_1) + u(B_2)$. Certains travaux considèrent plutôt le cas général, comme ceux de Plaut et Roughgarden [2018], tandis que d'autres, centrés sur le cas additifs, présentent tout de même quelques résultats pour d'autres cas (Caragiannis *et al.* [2019]). Notons que même les deux articles cités considèrent des fonctions d'utilité monotones (c'est-à-dire que si on a deux lots S et T tels que $S \subseteq T$, alors $u(S) \leq u(T)$). C'est le cas pour la très grande majorité des travaux dans le domaine, les principales exceptions étant ceux sur la répartition des corvées (où les objets d'utilité négative cassent la monotonie).

Exemple 1.1.1: Représentation des préférences

Considérons un agent a et un ensemble d'objets $\{h_1, h_2\}$. L'ensemble des lots possibles est $S = \{\emptyset, \{h_1\}, \{h_2\}, \{h_1, h_2\}\}$.

Supposons que a préfère $\{h_1, h_2\}$ à $\{h_1\}$, qu'il préfère $\{h_2\}$ à l'ensemble vide.

Le profil de préférences ordinaires pour a est :

$$a : \{h_1, h_2\} > \{h_1\} > \{h_2\} > \emptyset$$

Un profil de préférences cardinales pour a serait par exemple une fonction d'utilité u telle que :

- $u(\emptyset) = 0$
- $u(\{h_1\}) = 2$

- $u(\{h_2\}) = 1$
- $u(\{h_1, h_2\}) = 3$

Notons que dans cet exemple la fonction d'utilité u est additive car

$$\forall B_1, B_2 \in \{\emptyset, \{h_1\}, \{h_2\}\}, u(B_1 \cup B_2) = u(B_1) + u(B_2)$$

Si la forme des préférences sont compatibles avec toutes sortes de modèles, les autres choix sont structurels et vont décider à quelle catégorie le modèle appartient. La première de ces grandes différences sera ce qui va être alloué aux agents : soit il existe un ensemble de ressources qui sera alloué aux agents, soit les agents seront eux-mêmes la "ressource" et seront alloués entre eux.

1.1.1 Problème d'appariement

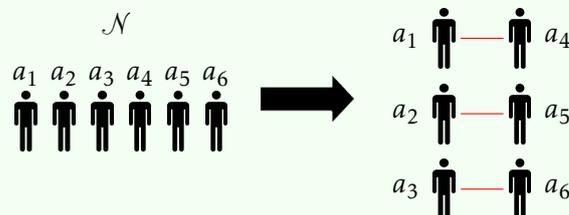
L'exemple type d'un problème d'appariement est le problème des colocataires, introduit par Gale et Shapley [1962]. Dans ce problème, on considère un ensemble d'agents \mathcal{N} de taille paire, et on cherche à associer les agents par deux pour former $\frac{|\mathcal{N}|}{2}$ couples de sorte à former une allocation ne contenant pas de "paire instable", c'est-à-dire deux agents non affectés qui, à cause de leur préférences, souhaitent se séparer de leurs paires respectives pour se mettre ensemble. Si il n'est pas garanti qu'une telle allocation existe dans le cas général, Bartholdi et Trick [1986] montrent cette existence pour certains types de préférences.

Exemple 1.1.2: Problème des colocataires

Considérons une instance avec un ensemble d'agents $\mathcal{N} = \{a_1, a_2, a_3, a_4, a_5, a_6\}$, et les profils de préférences (ordinaux) suivants :

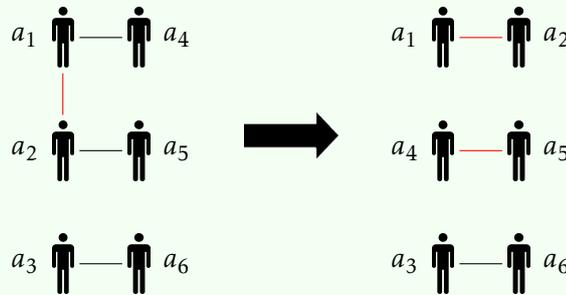
$$\begin{array}{ll} a_1 : a_2 > a_4 > a_3 > a_5 > a_6 & a_4 : a_1 > a_5 > a_3 > a_2 > a_6 \\ a_2 : a_1 > a_3 > a_4 > a_5 > a_6 & a_5 : a_2 > a_3 > a_4 > a_1 > a_6 \\ a_3 : a_6 > a_1 > a_4 > a_5 > a_2 & a_6 : a_3 > a_2 > a_4 > a_5 > a_1 \end{array}$$

Une allocation consiste à associer tous ces agents en paires.



Considérons par exemple l'allocation $(a_1 - a_4), (a_2 - a_5), (a_3 - a_6)$.

Nous remarquons cependant dans le profil de préférences que a_1 préfère a_2 à son colocataire (a_4) et que de son côté, a_2 préfère aussi a_1 à son colocataire (a_5). Par conséquent, la paire (a_1, a_2) est instable : a_1 et a_2 préféreront se mettre ensemble que rester avec leurs colocataires respectifs.



Nous nous retrouvons avec l'allocation $(a_1 - a_2), (a_4 - a_5), (a_3 - a_6)$, qui ne contient pas de paire instable, puisque a_1, a_2, a_3 , et a_6 sont avec leurs premiers choix et ne préfèrent donc aucun autre agent à leur colocataire actuel.

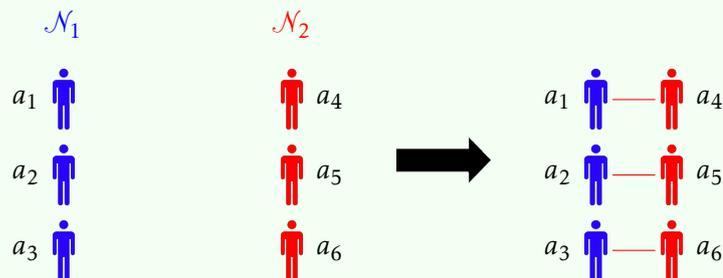
Le problème des mariages stables est un problème très proche dans lequel les agents sont séparés en deux ensembles différents, et l'on cherche à former des couples contenant un agent de chaque ensemble respectant la même règle de stabilité, c'est-à-dire ne contenant pas de paire d'agents provenant d'ensembles différents se préférant à leurs allocations respectives.

Exemple 1.1.3: Problème de mariage stable

Considérons une instance avec deux ensembles d'agents $\mathcal{N}_1 = \{a_1, a_2, a_3\}$ et $\mathcal{N}_2 = \{a_4, a_5, a_6\}$, et les profils de préférences (ordinaux) suivants :

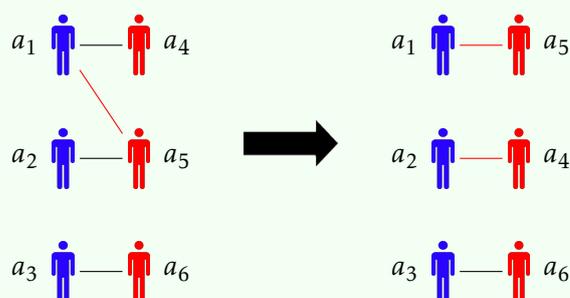
- | | |
|-------------------------|-------------------------|
| $a_1 : a_5 > a_4 > a_6$ | $a_4 : a_1 > a_3 > a_2$ |
| $a_2 : a_5 > a_4 > a_6$ | $a_5 : a_1 > a_3 > a_2$ |
| $a_3 : a_6 > a_4 > a_5$ | $a_6 : a_3 > a_2 > a_1$ |

Une allocation consiste à créer des paires contenant un élément de chaque ensemble.



Considérons par exemple l'allocation $(a_1 - a_4), (a_2 - a_5), (a_3 - a_6)$.

Cependant, la paire (a_1, a_5) est une paire contenant un agent de T_1 et un agent de T_2 qui se préfère l'un l'autre à leur conjoints respectifs (a_1 préfère a_5 à a_4 , et a_5 préfère a_1 à a_2). Cette paire est donc une paire instable, et rend l'allocation instable : ces deux agents préféreraient se mettre ensemble plutôt que se mettre avec leurs partenaires respectifs.



Nous nous retrouvons avec l'allocation $(a_1 - a_5), (a_5 - a_4), (a_3 - a_6)$, qui ne contient pas de paire instable, puisque a_1, a_5, a_3 , et a_6 sont avec leurs premiers choix et ne préfèrent donc aucun autre agent à leur conjoint actuel.

Il est facile de transformer une instance de ce problème en instance du problème des colocataires, cependant il existe pour ce problème une garantie d'existence d'allocation stable (ne contenant pas de paire instable) prouvée par Gale et Shapley [1962], qui proposent un algorithme générant une telle allocation. Cet algorithme est utilisé dans de nombreux problèmes réels, un exemple d'application est la plate-forme Parcoursup répartissant des places en universités et écoles aux étudiants post-bac.

Les jeux à coalitions hédoniques, aussi parfois simplement appelés jeux hédoniques (étudiés par exemples par Drèze et Greenberg [1980] et Bredereck *et al.* [2019]) sont une catégorie de problèmes où les agents vont chercher à former des coalitions de n'importe quelle taille, on y considèrera donc généralement que les agents sont alloués aux coalitions elles-mêmes plutôt qu'aux autres agents. Il existe plusieurs variantes des jeux hédoniques utilisées dans différents contextes, il est par exemple courant de considérer un ensemble de ressources à répartir entre les coalitions, comme le font Drèze et Greenberg [1980] et Bogomolnaia et Jackson [2002]). Une catégorie particulière de jeux hédoniques est celle des jeux *purement hédoniques*, c'est-à-dire les jeux où l'utilité des agents ne dépend que des identités des agents dans leur coalition. Un exemple d'un tel modèle est celui étudié par Bredereck *et al.* [2019], qui considèrent un jeu de diversité où l'ensemble des agents est partitionné en types et chaque agent cherche à appartenir à une coalition ayant une certaine proportion de chaque type. L'un des principaux objectifs lors de l'étude de ces problèmes est de s'assurer de la stabilité de la répartition des agents en coalitions. Il existe plusieurs définitions de stabilité : certaines individuelles, comme la Nash-stabilité (aucun agent n'a intérêt à quitter sa coalition pour en rejoindre

une autre), d'autre collectives, comme l'appartenance au cœur (il n'est pas possible de former une nouvelle coalition S en améliorant la situation de tous les agents de S).

Exemple 1.1.4: Jeu de diversité (exemple de jeu hédonique)

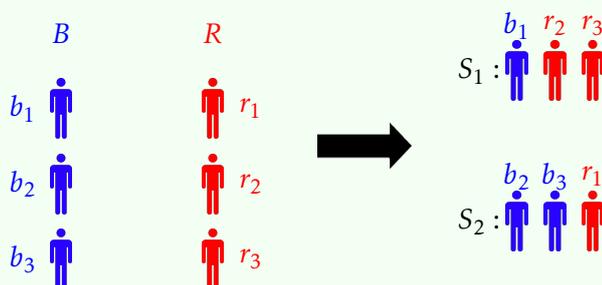
Nous considérons une instance du modèle décrit par Bredereck *et al.* [2019] constitué d'un ensemble d'agents \mathcal{N} divisé en deux types : le type rouge $R = \{r_1, r_2, r_3\}$ et le type bleu $B = \{b_1, b_2, b_3\}$.

Chaque agent a des préférences sur la proportion d'agents rouge dans sa coalition. Puisqu'il y a 6 agents dont 3 rouges, l'ensemble des proportions possibles est $\Theta = \{1, 0, \frac{1}{2}, \frac{1}{3}, \frac{2}{3}, \frac{1}{4}, \frac{3}{4}, \frac{2}{5}, \frac{3}{5}\}$ (notons que 0 est inaccessible aux agents rouges, et 1 inaccessible aux agents bleus).

Nous posons les profils de préférences des agents sur les proportions d'agents rouges :

$$\begin{array}{ll}
 r_1 : \frac{1}{3} > \frac{2}{5} > \frac{1}{2} > \frac{3}{5} > \frac{2}{3} > \frac{3}{4} > \frac{1}{4} > 1 & b_1 : \frac{2}{3} > \frac{3}{5} > \frac{1}{2} > \frac{2}{5} > \frac{1}{3} > \frac{1}{4} > \frac{3}{4} > 0 \\
 r_2 : \frac{1}{2} > \frac{1}{3} > \frac{2}{5} > \frac{2}{3} > \frac{1}{3} > \frac{1}{4} > \frac{3}{4} > 1 & b_2 : \frac{1}{2} > \frac{1}{3} > \frac{2}{5} > \frac{1}{3} > \frac{1}{4} > \frac{3}{4} > 0 \\
 r_3 : \frac{2}{3} > \frac{1}{2} > \frac{2}{5} > \frac{1}{3} > \frac{1}{4} > \frac{3}{4} > 1 & b_3 : \frac{1}{3} > \frac{2}{5} > \frac{1}{2} > \frac{3}{5} > \frac{2}{3} > \frac{3}{4} > \frac{1}{4} > 0
 \end{array}$$

Les agents vont alors former des coalitions. Nous chercherons généralement à générer des allocations stables, c'est-à-dire des allocations où aucun agent ne cherche à changer sa coalition.



Ici, les agents ont formé deux coalitions : $S_1 = \{b_1, r_1, r_2\}$, $S_2 = \{b_2, b_3, r_1\}$.

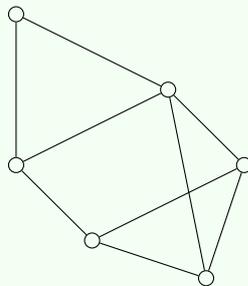
- cette allocation n'est pas Nash-stable : b_2 souhaiterait quitter S_2 (proportion de $\frac{1}{3}$) pour rejoindre S_1 (proportion de $\frac{1}{2}$).
- cette allocation n'est pas dans le cœur : b_2 et r_2 ont intérêt à quitter leur coalition respective (de proportion $\frac{1}{3}$ et $\frac{2}{3}$ respectivement) pour former une nouvelle coalition $S = \{b_2, r_2\}$ (de proportion $\frac{1}{2}$).

Les **jeux de Schelling** (introduits par Schelling [1969]) constituent une catégorie de problèmes qui cherchent également à lier les agents au moyen d'une affectation, mais cette fois de manière indirecte. Ces problèmes se composent d'un ensemble d'agents et d'un graphe, on affectera alors les agents aux sommets du graphe (un agent maxi-

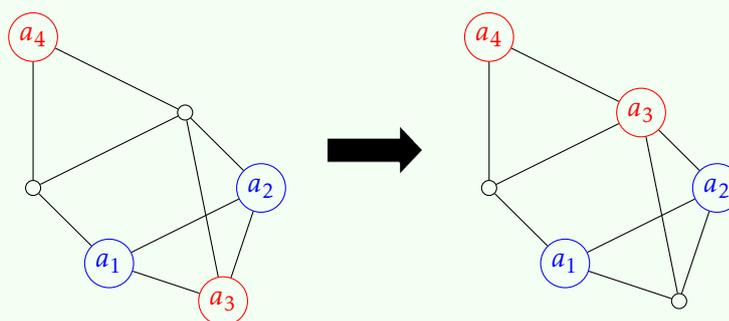
mum par sommet, un sommet pour chaque agent), en considérant que chaque agent influence les agents alloués aux sommets reliés au sien par une arête. Les variantes de ce problème se différencient en fonction de quelle dynamique est appliquée aux agents (peuvent-ils sauter sur un sommet vide, ou échanger leurs places entre eux), et de quels sont les objectifs des agents (les agents vont typiquement chercher à obtenir un sommet précis ou à se regrouper par type). Les recherches sur ces problèmes étudient généralement l'existence d'équilibre (par exemple Chauhan *et al.* [2018] et Elkind *et al.* [2019]).

Exemple 1.1.5: Jeux de Schelling

Considérons une instance avec un ensemble d'agents $\mathcal{N} = \{a_1, a_2, a_3, a_4\}$ partitionné en deux types $T_1 = \{a_1, a_2\}$ et $T_2 = \{a_3, a_4\}$, et d'un graphe $G = (V, E)$ de la forme suivante :



Les agents vont se répartir sur les sommets du graphe, et pourront par la suite se déplacer sur les sommets vides. Leur objectif sera de faire en sorte qu'au moins la moitié de leurs voisins soient du même type qu'eux (nous supposons que les agents isolés sont satisfaits).



Dans l'allocation ci-dessus à gauche, l'agent a_3 n'est pas satisfait (aucun de ses voisins n'est de son type), il se déplace donc sur un nouveau sommet. Dans la nouvelle allocation (à droite), tous les agents sont satisfaits.

1.1.2 Ressources divisibles et indivisibles

Dans la suite de ce manuscrit, nous nous concentrerons sur les problèmes d'allocation de ressources, c'est-à-dire que nous chercherons toujours à allouer un ensemble de ressources \mathcal{M} à un ensemble d'agents \mathcal{N} . Nous nous concentrerons sur une ensemble de ressources indivisibles, mais il nous faut remarquer qu'il existe également une littérature très étendue pour le cas où les ressources sont divisibles.

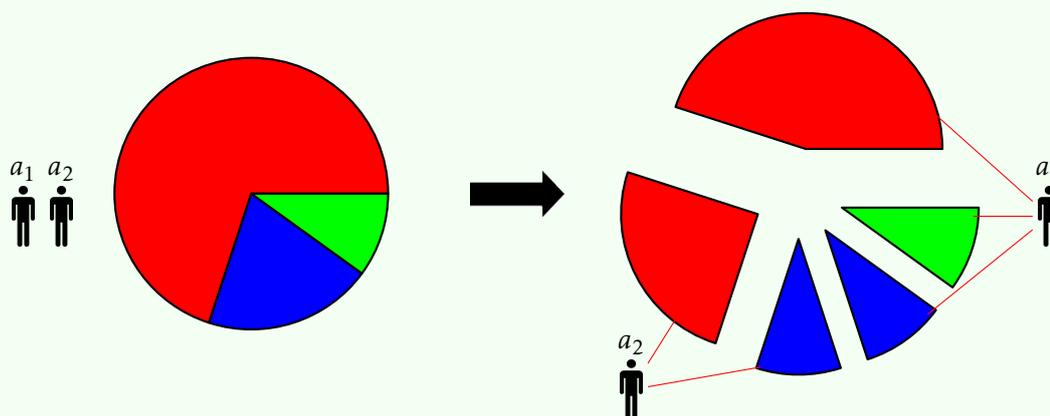
Dans le cas le plus simple, il existe une seule ressource divisible homogène, comme dans le problème de Banqueroute (discuté par Thomson [2003]), où une somme d'argent doit être divisée entre un certain nombre de créanciers, à qui il est dû une somme différente pour chacun. Dans le problème de la découpe de gâteau, on cherche également à diviser une ressource entre plusieurs agents, mais cette ressource est hétérogène : elle peut être composé de plusieurs segments qui peuvent avoir valeurs différentes pour chaque agent. Une étude générale du problème a été effectuée par Brams et Taylor [1996], qui proposent une sélection d'algorithmes générant des allocations satisfaisant divers critères d'équité.

Exemple 1.1.6: Partage de gâteau

Considérons une instance avec un ensemble d'agents $\mathcal{N} = \{a_1, a_2\}$ et d'une ressource divisible constituée de trois segments distincts (rouge, bleu et vert). Nous supposons les profils d'utilité de la manière suivante :

	rouge	bleu	vert
a_1	7	2	1
a_2	14	2	0

Chaque segment peut être séparé de manière arbitraire avant d'être alloué aux agents.



Ici, notre allocation est telle que :

- a_1 reçoit $\frac{9}{14}$ du segment rouge, $\frac{1}{2}$ du segment bleu et le segment vert en entier. Son utilité est donc de :

$$u_{a_1} = \frac{9}{14} \cdot 7 + \frac{1}{2} \cdot 2 + 1 = 6,5$$

- a_2 reçoit $\frac{5}{14}$ du segment rouge et $\frac{1}{2}$ du segment bleu. Son utilité est donc de :

$$u_{a_2} = \frac{5}{14} \cdot 14 + \frac{1}{2} \cdot 2 = 6$$

À mi-chemin entre les modèles avec ressources divisibles et indivisible, il existe des modèles où les ressources sont indivisibles mais mises en commun pour un groupe d'agents. Par exemple, dans le problème étudié par Kyropoulou *et al.* [2020] et Segal-Halevi et Suksompong [2019], l'ensemble des agents est partitionné en un certain nombre de types, et les ressources sont allouées à ces types. On considère alors qu'une ressource allouée à un type donné est partagée entre tous les agents de ce type. Les agents ont des préférences sur les lots d'objets alloués à leur type, et les travaux mentionnés étudient comment transcrire la notion d'équité à ce cadre.

Une autre possibilité de compromis entre les deux cas est le modèle d'allocation de ressources indivisibles avec transfert d'argent, conçue pour contourner les difficultés propre au cas des ressources indivisibles en rajoutant une ressource divisible transférable entre agents, l'argent, qui permet une plus grande flexibilité d'allocation. Des résultats généraux sur ces problèmes sont présentés dans les travaux de Roth et Sotomayor [1990]. Un exemple particulier d'un problème de ce type est le problème de la division du loyer. Dans ce problème, un certain nombre de colocataires doivent se répartir les chambres d'un logement, puis décider de comment partager le loyer. Les chambres ne peuvent pas être partagées, un colocataire ne peut en obtenir qu'une seule, mais le loyer est divisible et permet donc d'équilibrer les parts de chacun. Les travaux de Gal *et al.* [2016] constituent un exemple d'étude récente de ce problème.

Exemple 1.1.7: Problème de la division du loyer

Considérons une instance constituée d'un ensemble d'agents $\mathcal{N} = \{a_1, a_2, a_3\}$, un ensemble d'objets $\mathcal{M} = \{h_1, h_2, h_3\}$ et un loyer total $R = 1000$. Nous posons la matrice des utilités des objets pour les agents :

U^I	h_1	h_2	h_3
a_1	600	400	200
a_2	150	500	300
a_3	200	100	350

L'utilité totale d'un agent sera l'utilité qu'il reçoit de l'objet moins la valeur de

la part du loyer qui lui est attribuée. L'utilité de l'agent a recevant l'objet h et la part de loyer r_a sera donc : $u_a = U_{a,h}^I - r_a$

Chaque agent se voit allouer un objet, puis une part du loyer. Ici :

- a_1 reçoit h_1 et une part du loyer égale à 400, son utilité est donc $u_{a_1} = 600 - 400 = 200$
- a_2 reçoit h_2 et une part du loyer égale à 350, son utilité est donc $u_{a_2} = 500 - 350 = 150$
- a_3 reçoit h_3 et une part du loyer égale à 250, son utilité est donc $u_{a_3} = 350 - 250 = 100$

Dans un problème d'allocation de ressources indivisibles classique, chaque ressource ne peut être allouée qu'à un seul agent. Cependant, il reste plusieurs caractéristiques importantes dont les différences engendrent des modèles distincts. La première de ces différences est la question de l'éventuelle réciprocité des préférences. En effet, dans certains problèmes, les ressources sont possédées par des personnes ou organismes qui auront des préférences sur quels agents se verront alloué leurs ressources, tandis que dans d'autres, seuls les agents auront des préférences. Le deuxième cas, plus courant, est appelé "one-sided", tandis que le premier cas est appelé "two-sided", et peut être vu comme une généralisation du problème du mariage stable, puisque nous aurons deux ensembles, chacun ayant des préférences l'un par rapport à l'autre, que nous chercherons à allouer entre eux. Cependant, l'ensemble de ressources ne pourra pas toujours être considéré comme un second ensemble d'agents. Un exemple de problème d'allocation de ressources "two-sided" est le problème du choix des écoles, consistant à répartir des places en écoles entre des étudiants.

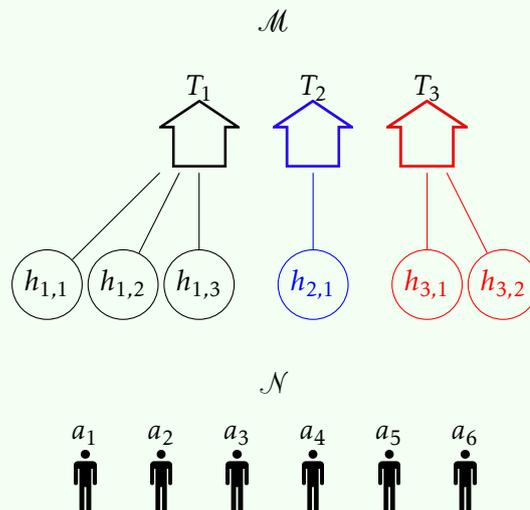
Le problème du choix des écoles est constitué d'un ensemble d'étudiant et d'un ensemble de places en école (aussi appelés contrats), chacun associé à une école particulière. Les étudiants ont des préférences sur les écoles, et parfois également entre les différentes places d'une même école, tandis que les écoles ont des préférences sur les étudiants. On cherche alors une allocation sans paire (étudiant, contrat) instable

pour la même définition de paire instable que pour le problème du mariage stable (on parlera alors d'*envie justifiée*). Gale et Shapley [1962] ont montré qu'une telle allocation existe toujours, grâce à une adaptation de leur algorithme pour le mariage stable permettant de générer une telle allocation. Ce problème a été étudié dans l'objectif d'imposer d'autres contraintes à l'allocation finale, par exemple par Goto *et al.* [2015] et Aziz et Sun [2021].

Exemple 1.1.8: Problème du choix des écoles

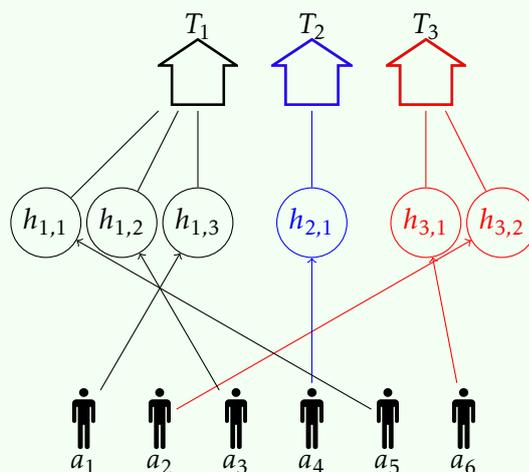
Considérons une instance constituée d'un ensemble d'agents $\mathcal{N} = \{a_1, a_2, a_3, a_4, a_5, a_6\}$ et d'un ensemble d'objets (contrats) \mathcal{M} partitionné en trois types (écoles) $T_1 = \{h_{1,1}, h_{1,2}, h_{1,3}\}$, $T_2 = \{h_{2,1}\}$ et $T_3 = \{h_{3,1}, h_{3,2}\}$. Les préférences se feront avec les écoles plutôt que les contrats. Nous posons les profils de préférences :

- | | |
|-------------------------|---|
| $a_1 : T_1 > T_2 > T_3$ | |
| $a_2 : T_2 > T_1 > T_3$ | |
| $a_3 : T_1 > T_3 > T_2$ | $T_1 : a_1 > a_2 < a_3 > a_4 > a_5 > a_6$ |
| $a_4 : T_2 > T_3 > T_1$ | $T_2 : a_4 > a_6 < a_1 > a_2 > a_5 > a_3$ |
| $a_5 : T_1 > T_3 > T_2$ | $T_3 : a_6 > a_5 > a_3 > a_2 > a_1 > a_4$ |
| $a_6 : T_3 > T_2 > T_1$ | |



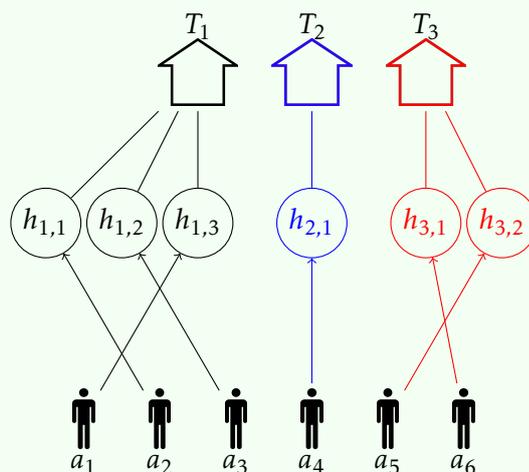
Ce problème est similaire à celui du mariage stable : nous allons associer les contrats et les agents deux à deux, en tenant compte des préférences des agents et des écoles.

Considérons l'allocation suivante :



Dans cette allocation, a_2 est alloué à un objet de T_3 mais lui préfère T_1 , tandis que T_1 préfère a_2 à certains des agents qui lui sont alloués (a_3 et a_5). Cette allocation contient donc de l'envie justifiée.

Une manière de supprimer cette envie est d'échanger les allocations de a_2 et a_5 . Nous obtiendrons alors l'allocation suivante :



Les travaux présentés dans ce manuscrit se concentreront sur le problème de *house allocation*, étudié par exemple par Zhou [1990] et Abdulkadiroğlu et Sönmez [1998]. Ce modèle est un problème d'allocation de ressources indivisibles "one-sided" où l'on rajoute la contrainte que chaque agent ne doit être alloué qu'à une seule ressource.

1.1.3 Types d'agents

Une variante qui servira de base de départ aux travaux présentés dans ce manuscrit est celle d'allocation à des groupes d'agents. Dans ces modèles, l'ensemble des agents est

partitionnés en un certain nombre de types disjoints. Des exemples de cette catégorie de modèle que nous avons déjà mentionnés sont les problèmes où les ressources sont allouées à un type dont les agents se partageront la ressource. Une autre utilisation de ce concept concerne la représentation de la diversité, que l'on étudiera en détail plus tard. Dans ces problèmes, on cherche généralement à appliquer aux types des notions d'équité normalement utilisées à l'échelle des agents. Un problème de ce genre est considéré par Benabbou *et al.* [2019]. Une manière de pousser ce concept plus loin est de partitionner également l'ensemble des ressources en un ensemble de blocs de ressources. Cette modification permet à Benabbou *et al.* [2018] de considérer des notions telles que les quotas de diversité entre types et blocs (dont nous reparlerons dans la suite).

Notons qu'une telle notion n'est pas exclusive au problème de *house allocation*, une modification similaire est également souvent appliquée au problème du choix des écoles, pour représenter l'origine diverse des étudiants (par exemple par Echenique et Yenmez [2015]). Il nous faut de plus noter que dans ces problèmes, il existe déjà une partition de l'ensemble des contrats, puisque chacun d'eux est associé à une école. Certains problèmes de choix d'écoles se démarquent en permettant aux agents de faire partie de plusieurs types en même temps, l'ensemble des types ne forme alors plus une partition, mais simplement une famille de sous-ensembles couvrant l'ensemble des agents (un tel modèle est étudié par Aziz *et al.* [2020]).

1.2 Qualité des solutions

1.2.1 Notion d'efficacité

Dans la majorité de la littérature, les deux critères utilisés pour mesurer la qualité des solutions sont l'efficacité et l'équité. La première, l'efficacité, est la plus simple à concevoir : on souhaite que les agents obtiennent une allocation qui les satisfasse autant que possible. Pour des préférences cardinales, cela veut dire que l'utilité qu'ils ont pour leur allocation est aussi grande que possible, tandis que pour des utilités ordinales, cela veut dire que leur allocation doit être aussi haute que possible dans leur ordre de préférences. Notons que dans ce dernier cas, il est toujours possible de se ramener au cas des préférences cardinales en utilisant des méthodes telles que les scores de Borda (Emerson [2013]).

Un critère naturel permettant de caractériser les allocations "efficaces" est la maximisation du **bien-être social utilitariste** (Moulin [1988]), c'est-à-dire de la somme des utilités des agents. La difficulté de calcul d'une telle allocation dépend du modèle, elle est triviale dans le cas des problèmes d'allocations de ressource indivisible classiques (il suffit d'allouer chaque ressource à un des agents qui en tire le plus d'utilité), tandis qu'il est possible de les calculer en temps polynomial pour le problème de *house allocation*.

Définition 1.2.1: Bien-être social utilitariste

Soit une instance I d'un problème d'allocation de ressources contenant un ensemble d'agents \mathcal{N} et un ensemble d'objets \mathcal{M} , tels que tout agent $a \in \mathcal{N}$ dispose d'une fonction d'utilité u_a sur les sous-ensembles de \mathcal{M} .

Le **bien-être social utilitariste** d'une allocation A sur I est la valeur :

$$sw_U(A) = \sum_{a \in \mathcal{N}} u_a(A(a))$$

Plus rarement, on considèrera le **bien-être social égalitariste**, égale à l'utilité de l'agent le moins satisfait. Cette mesure peut être maximisée pour garantir une certaine équité entre les agents, un concept que nous verrons plus en détail dans la suite.

Définition 1.2.2: Bien-être social égalitariste

Soit une instance I définie comme dans la définition 1.2.1.

Le **bien-être social égalitariste** d'une allocation A sur I est la valeur :

$$sw_{Eq}(A) = \min_{a \in \mathcal{N}} u_a(A(a))$$

Une autre notion parfois utilisée est celle du **bien-être de Nash**, qui consiste à multiplier les utilités des agents entre elles. Comme nous le verrons dans la suite, cette notion est très compatible avec certaines notions d'équité.

Définition 1.2.3: Bien-être de Nash

Soit une instance I définie comme dans la définition 1.2.1.

Le **bien-être de Nash** d'une allocation A sur I est la valeur :

$$NW(A) = \prod_{a \in \mathcal{N}} u_a(A(a))$$

Si l'on cherchera parfois les allocations maximisant ces valeurs, il sera souvent plus utile de caractériser une catégorie d'allocations "efficaces". Pour cela, on utilisera un critère plus général, comme les **optimums de Pareto**.

Une allocation A_1 est dite **Pareto-dominée** par une allocation A_2 si et seulement si aucun agent ne préfère (strictement) son lot dans A_1 à son lot dans A_2 , et au moins un agent préfère (strictement) son lot dans A_2 à son lot dans A_1 . Une allocation est alors **Pareto-optimale** si elle n'est Pareto-dominée par aucune autre allocation. Dans le cadre de préférences cardinales, on peut utiliser la définition suivante :

Définition 1.2.4: Allocation pareto-optimale

Soit une instance I définie comme dans la définition 1.2.1. Une allocation A_1 sur I est dite **Pareto-optimale** si et seulement si il n'existe pas d'allocation A_2 sur I vérifiant les deux conditions suivantes :

- $\forall a \in \mathcal{N}, u_a(A_1(a)) \leq u_a(A_2(a))$
- $\exists a \in \mathcal{N}, u_a(A_1(a)) < u_a(A_2(a))$

Il est trivial de montrer que toute allocation maximisant le bien-être social utilitariste est Pareto-optimale. En ce qui concerne le bien-être de Nash, il est possible de trouver un résultat similaire, en traitant à part le cas où le bien-être maximum est de zéro (voir Caragiannis *et al.* [2019]). Enfin, parmi les allocations qui maximisent le bien-être social égalitariste, au moins une est Pareto-optimale.

Les travaux de Caragiannis *et al.* [2019] et de Halpern *et al.* [2020] entre autres utilisent cette propriété du bien-être de Nash pour garantir l'efficacité de leurs allocation pendant qu'ils se concentrent sur d'autres aspects.

Une notion importante en lien avec la notion d'optimum de Pareto est la notion de **gâchis**, considérée par Conitzer *et al.* [2019]. Une ressource est dite **gâchée** si la retirer ne diminue pas l'utilité de l'agent auquel elle est alloué (ou si elle n'est alloué à aucun agent), et l'ajouter à un autre agent augmente l'utilité de ce dernier. Une allocation est dite **sans gâchis** si aucune ressource n'est gâchée.

Définition 1.2.5: Gâchis

Soit une instance I définie comme dans la définition 1.2.1.

Un objet $h \in \mathcal{M}$ est **gâché** dans l'allocation A si il peut augmenter l'utilité d'un agent ($\exists a \in \mathcal{N}, u_a(A(a)) < u_a(A(a) \setminus \{h\})$) et il satisfait l'une des deux conditions suivantes :

- il n'est alloué à personne ($\forall a \in \mathcal{N}, h \notin A(a)$)
- il est alloué à un agent a' tel que l'utilité de a' ne changerait pas si ce dernier venait à perdre l'objet h ($\exists a' \in \mathcal{N}, h \in A(a')$ et $u_{a'}(A(a')) = u_{a'}(A(a') \setminus \{h\})$)

Si aucun objet n'est gâché, l'allocation est dite *sans gâchis*

Cette notion a été adaptée au contexte où les agents sont séparés en types par Benabbou *et al.* [2019]. De toute évidence, une allocation Pareto-optimale est sans gâchis, mais la réciproque n'est pas vraie. L'absence de gâchis est donc une propriété plus facile à remplir que la Pareto-optimalité.

Montrons la manière dont ces notions interagissent entre elles sur un exemple :

Exemple 1.2.1: Différentes notions d'efficacité

Soit une instance du problème de *house allocation* constitué d'un ensemble d'agents $\mathcal{N} = \{a_1, a_2, a_3\}$ et d'un ensemble d'objets $\mathcal{M} = \{h_1, h_2, h_3\}$. Supposons que les fonctions d'utilité soient définies par le tableau suivant :

	h_1	h_2	h_3
a_1	5	4	7
a_2	1	4	0
a_3	2	0	5

Dans ce problème, h_2 sera gâché si il est alloué à a_3 , et h_3 sera gâché si il est alloué à a_2 . Par conséquent, il existe 3 allocations sans gâchis :

- l'allocation A_1 telle que $A_1(a_1) = h_1$, $A_1(a_2) = h_2$, et $A_1(a_3) = h_3$, de vecteur d'utilité (5, 4, 5)
- l'allocation A_2 telle que $A_2(a_1) = h_3$, $A_2(a_2) = h_2$, et $A_2(a_3) = h_1$, de vecteur d'utilité (7, 4, 2)
- l'allocation A_3 telle que $A_3(a_1) = h_2$, $A_3(a_2) = h_1$, et $A_3(a_3) = h_3$, de vecteur d'utilité (4, 1, 5)

Lorsque l'on compare les vecteurs d'utilité, on remarque que celui de A_3 est Pareto-dominé par celui de A_1 , car $4 < 5$ (pour a_1), $1 < 4$ (pour a_2) et $5 \leq 5$ (pour a_3). Ni A_1 ni A_2 ne Pareto-domine l'autre puisque $5 < 7$ (pour a_1) mais $5 > 2$ (pour a_3). Par conséquent, l'ensemble des allocations Pareto-optimale est $\{A_1, A_2\}$.

Parmi ces deux allocations, A_1 a un bien-être social utilitariste de $sw(A_1) = 5 + 4 + 5 = 14$ tandis que A_2 en a un de $sw(A_2) = 7 + 4 + 2 = 13$, l'allocation maximisant le bien-être social utilitariste est A_1 .

Certains modèles n'allouent pas leur biens de manière déterministe, mais considèrent des loteries où l'allocation est tirée au sort, et utilisent donc des mécanismes qui génèrent des distributions de probabilités sur les allocations plutôt que des allocations. Dans ces modèles, il peut être utile de considérer l'efficacité avant et après tirage. Mesurer l'efficacité avant tirage consiste à remplacer l'utilité du lot d'un agent par son utilité de gain, tandis que la mesurer après tirage consiste à considérer toutes les répartitions qui peuvent effectivement être retournées par le tirage. On dira ainsi qu'un mécanisme garantit la Pareto-optimalité *ex-ante* si le vecteur des espérances de gain des agents pour la distribution de probabilité retournée par le mécanisme est Pareto-optimal, et qu'il garantit la Pareto-optimalité *ex-post* si toutes allocations qui peuvent être tirées sont Pareto-optimales. L'idée de vérifier l'optimalité avant et après tirage est ancienne, mais les termes Pareto-optimal *ex-ante* et Pareto-optimal *ex-post* ont été codifiés par Dreze [1970]. Notons que toute allocation qui est Pareto-optimale *ex-ante* est nécessairement Pareto-optimale *ex-post*.

1.2.2 Notions d'équité

La notion d'équité est complexe à définir, par conséquent, de nombreuses mesures différentes ont vu le jour. La méthode la plus simple est sans doute la maximisation du bien-être social égalitariste, que nous avons vu dans la partie précédente, mais cette notion admet un petit nombre de solutions. On préférera généralement qualifier tout une catégorie de solution comme "équitable", par conséquent la plupart des méthodes reposent sur des propriétés dont la satisfaction prouve l'équité. Nous allons donc nous intéresser à divers propriétés d'équité et aux manières de garantir leur satisfaction.

La notion de **proportionnalité**, introduite par Steinhaus [1948], formalise l'idée que, si l'ensemble des agents \mathcal{N} contient $n \in \mathbb{N}$ agents, alors chaque agent devrait recevoir une part qu'il estime valoir au moins $\frac{1}{n}$ de la valeur de l'ensemble.

Définition 1.2.6: Proportionnalité

Une allocation A vérifie la **proportionnalité** si :

$$\forall a \in \mathcal{N}, u_a(A_a) \geq \frac{u_a(\mathcal{M})}{|\mathcal{N}|}$$

où pour un agent a , u_a désigne sa fonction d'utilité, et A_a son lot d'objets alloué dans A .

La notion de **maximin share** étend le concept de proportionnalité pour le rendre mieux adapté aux cas des ressources indivisibles. Introduite par Budish [2011], l'idée de cette mesure est la suivante : si un agent pouvait partitionner l'ensemble des ressources comme il le souhaite en autant de lots qu'il y a d'agents, mais devait en contrepartie choisir son lot en dernier (et risquerait donc de se retrouver avec le pire lot), quelle serait l'utilité minimum dont il pourrait s'assurer? Dans le cas où les préférences sont cardinales, le **maximin share** se formalise donc comme le maximum sur l'ensemble des allocations, du minimum sur les lots de l'allocation, de l'utilité du lot pour l'agent.

Définition 1.2.7: Maximin share

Une allocation A vérifie **maximin share** si et seulement si :

$$\forall a_1 \in \mathcal{N}, u_{a_1}(A_{a_1}) \geq \max_{A^* \in \Pi_n(\mathcal{M})} \min_{a_2 \in \mathcal{N}} u_{a_2}(A_{a_2}^*)$$

où pour un agent a , u_a désigne sa fonction d'utilité, et A_a son lot d'objets alloué dans A , et $\Pi_n(\mathcal{M})$ désigne l'ensemble des partitions de \mathcal{M} en $n = |\mathcal{N}|$ ensembles.

Une variante appelée **pairwise maximin share**, introduite par Caragiannis *et al.* [2019], requiert que cette condition soit également vraie sur toute réduction de l'instance à une paire d'agents, c'est-à-dire que pour toute paire d'agents, chaque agent doit estimer avoir au moins ce qu'il pourrait obtenir si il coupait en deux leurs allocations

combinées et choisissait la pire moitié. Une autre variante appelée **Groupwise maximin share** a été introduite par Barman *et al.* [2018], et étend le concept de pairwise maximin share aux groupes d'agents de taille supérieure à deux. Cette variante est une propriété plus forte qui impose que pour tout sous-ensemble de l'ensemble d'agents, l'instance limitée à ce sous-ensemble et à leurs allocations vérifie aussi maximin share.

La plupart des notions d'équité présentées jusqu'à maintenant ne demandait aux agents que de considérer leur propre allocation et l'ensemble de ressources. Cependant, la notion suivante, **l'envie**, introduit par Foley [1967], leur demande de comparer leur allocation à celles des autres agents. On dit qu'un agent est **envieux** si il préfère l'allocation d'un autre agent à la sienne. On dit qu'une allocation est **sans-envie** si et seulement si aucun agent n'est envieux. Cette définition fonctionne pour des préférences cardinales comme pour des préférences ordinales. Dans le cadre de préférences cardinales, on peut formaliser la définition du caractère sans-envie comme suit :

Définition 1.2.8: Allocation sans-envie

Une allocation A est dite **sans-envie** si et seulement si :

$$\forall i_1, i_2 \in \mathcal{N}, u_{i_1}(A_{i_1}) \geq u_{i_1}(A_{i_2})$$

où pour un agent i , u_i désigne sa fonction d'utilité, et A_i son lot d'objet alloué dans A .

De toute évidence, une allocation à la fois **sans-envie** et **sans-gâchis** n'est pas toujours possible : par exemple, si il y a plusieurs agents et un seul objet, les agents qui n'obtiennent pas l'objet seront envieux de celui qui le reçoit. Par conséquent, une relaxation de la notion a été proposée par Budish [2011], la propriété **sans-envie à un bien près**. Cette relaxation modifie la définition pour autoriser un agent à en envier un autre, du moment qu'il est possible d'annuler cette envie en supprimant un objet du lot alloué à j .

Définition 1.2.9: Allocation sans-envie à un bien près

Une allocation A est dite **sans-envie à un bien près** si et seulement si :

$$\forall i_1, i_2 \in \mathcal{N}, \exists h \in A_{i_2}, u_{i_1}(A_{i_1}) \geq u_{i_1}(A_{i_2} \setminus h)$$

où pour un agent i , u_i désigne sa fonction d'utilité, et A_i son lot d'objet alloué dans A .

Il a été montré par Caragiannis *et al.* [2019] que pour le problème d'allocation de ressources indivisibles classique, il existe toujours une allocation sans-envie à un bien près. Une deuxième relaxation de l'envie a été introduite par Caragiannis *et al.* [2019] (inspiré des travaux de Gourvès *et al.* [2014], qui apporte un résultat similaire sur les

matroïdes). Cette notion, appelée **sans envie à n'importe quel bien près**, modifie la définition d'origine du caractère sans-envie pour autoriser un agent à en envier un autre, du moment qu'il est possible d'annuler cette envie en supprimant n'importe quel objet du lot alloué à j .

Définition 1.2.10: Allocation sans-envie à n'importe quel bien près

Une allocation A est dite **sans-envie à n'importe quel bien près** si et seulement si :

$$\forall i_1, i_2 \in \mathcal{N}, \forall h \in A_{i_2}, u_{i_1}(A_{i_1}) \geq u_{i_1}(A_{i_2} \setminus h)$$

où pour un agent i , u_i désigne sa fonction d'utilité, et A_i son lot d'objet alloué dans A .

Cette propriété est plus forte que la propriété sans-envie à un bien près, en restant plus faible que la propriété sans-envie, et peut donc servir d'intermédiaire entre les deux. L'existence d'une allocation EFX dans le cas général est une question ouverte, cependant il existe déjà des résultats d'existence sur certains cas particuliers : instances à deux agents (Plaut et Roughgarden [2018]), instances à 3 agents avec fonctions d'utilité additives (Chaudhury *et al.* [2024]), instances où toutes les utilités marginales des objets sont binaires (Bu *et al.* [2023]).

Il nous faut noter l'existence de plusieurs travaux considérant les relations entre les diverses notions d'équité. Un exemple d'un tel article est celui de Aziz et Rey [2020], qui en introduisant ses propres notions les place en relation avec les notions déjà existantes. Les travaux de Bouveret et Lemaître [2015] se concentrent sur cet aspect et construisent une échelle des notions d'équité contenant entre autres les notions présentées ici. Parmi les résultats qu'ils présentent, ils démontrent en particulier que toutes allocations sans envie vérifie le maximin share, et que toute allocation vérifiant le maximin share vérifie aussi la proportionnalité.

De la même manière que pour les notions d'efficacité, les notions d'équité peuvent être adaptées au cas des allocations aléatoires. Ainsi, Gajdos et Tallon [2002] introduit les notions de **sans-envie ex-ante** et **sans-envie ex-post** avec les définitions suivantes : une distribution sur les allocations est sans-envie ex-ante si aucun agent n'a une espérance pour un lot d'un autre agent qui soit supérieure à l'espérance qu'il a pour son lot, et elle est sans-envie ex-post si toutes les allocations qu'elle peut tirer sont sans-envie. Notons qu'à l'inverse de l'efficacité, si une allocation est sans-envie ex-post, alors elle est nécessairement sans-envie ex-ante (résultat prouvé par Gajdos et Tallon [2002]).

Plusieurs travaux ont tenté d'adapter la définition d'envie au cadre de groupes d'agents, chacune sur un modèle différent. Benabbou *et al.* [2019] considèrent un problème de house allocation où les agents sont partitionnés en types, et chaque type calcule son utilité comme la somme des utilités des agents qui le composent, cette valeur est ensuite utilisée pour déterminer si un type en envie un autre. Segal-Halevi et Suksompong [2019] considèrent un modèle où les agents sont partitionnés en types,

les objets sont alors alloués à ces types et partagés entre les agents d'un même type. Les agents d'un type décident alors démocratiquement si le type est envieux ou non, en fonction de leur propre fonction d'utilité. Enfin, Aleksandrov et Walsh [2018b] considèrent une variante d'envie permettant à deux groupes d'agents de se comparer entre eux, et cherchent des allocations qui soient sans envie pour toute paire de groupes d'agents.

1.2.3 Mécanismes optimisant l'équité

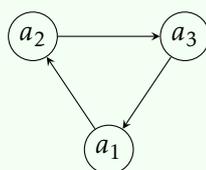
Les propriétés d'équité étant très diverses, les méthodes visant générer des allocations les satisfaisant le sont également.

La méthode d'élimination des cycles d'envie, introduite par Lipton *et al.* [2004], se concentre sur la notion d'envie, et représente l'allocation sous forme de graphe orienté où les agents représentent les sommets, et le sommet d'un agent i est lié au sommet d'un agent j si et seulement si i est envieux de j . Si il existe un cycle dans le graphe, nous pouvons alors échanger les allocations en remontant les arêtes et améliorer la situation de tout le monde. Cette méthode garantie de trouver une allocation sans-envie à un bien près si les préférences sont monotones. Elle a de plus été utilisée par Plaut et Roughgarden [2018] pour la génération d'allocations sans-envie à n'importe quel bien près.

Exemple 1.2.2: Principe d'élimination par des cycles d'envie

Supposons un problème d'allocation contenant 3 agents différents a_1 , a_2 et a_3 . Soit l'allocation A telle que :

- $u_{a_1}(A(a_1)) < u_{a_1}(A(a_2))$ (a_1 envie a_2)
- $u_{a_2}(A(a_2)) < u_{a_2}(A(a_3))$ (a_2 envie a_3)
- $u_{a_3}(A(a_3)) < u_{a_3}(A(a_1))$ (a_3 envie a_1)



Dans cette situation, on peut créer une nouvelle allocation A' en échangeant les lots d'objets de manière circulaire : a_1 reçoit le lot de a_2 ($A'(a_1) = A(a_2)$), a_2 reçoit le lot de a_3 ($A'(a_2) = A(a_3)$), et a_3 reçoit le lot de a_1 ($A'(a_3) = A(a_1)$). L'allocation A est Pareto-dominée par l'allocation A' . De plus, le graphe d'envie de A' ne comporte plus aucune arête, puisque les envies du cycle ont disparu.

En ce qui concerne la notion de *maximin share*, Ghodsi *et al.* [2018] proposent

l'algorithme de *bag filling*, qui garantit de retourner une allocation respectant une approximation de maximin share, le degré d'approximation dépendant des hypothèses sur les fonctions d'utilité. Cet algorithme consiste à remplir un sac d'objets que l'on "vide" régulièrement en allouant son contenu à un agent.

1.2.4 Mécanismes optimisant efficacité et équité

Efficacité et équité ne sont pas contradictoires, mais il peut être difficile de trouver des solutions conciliant les deux notions de manière satisfaisante. Par conséquent, de nombreux travaux se sont penchés sur la création de méthodes et algorithmes permettant de créer des allocations satisfaisant des propriétés d'efficacité et d'équité.

Un exemple de méthode pertinente pour les deux notions est la maximisation du bien-être de Nash. Il a été montré par Caragiannis *et al.* [2019] que l'allocation maximisant cette mesure est à la fois un optimum de Pareto, et sans-envie à un bien près, dans le cadre de fonctions d'utilité additives. Les travaux de Halpern *et al.* [2020] montrent de plus que cette méthode est résistante à la manipulation dans le cadre de fonctions d'utilité binaires. Caragiannis *et al.* [2019] discutent cependant de la difficulté de calculer ce maximum, NP-difficile dans le cas général, et proposent donc une manière de calcul en construisant un programme linéaire en nombre entier qui soit suffisamment rapide à résoudre. De leur côté, Halpern *et al.* [2020], dans le cas restreint des utilités binaires, montrent que ce maximum est calculable en temps polynomial. Autre travaux notable, Camacho *et al.* [2023] se placent dans un contexte où toute allocation qui maximise du bien-être de Nash maximise aussi le bien-être social utilitariste, puis proposent un algorithme capable d'approcher ces allocations en générant des allocations sans-envie à un bien-près (ou à n'importe quel bien près) qui maximisent le bien-être social utilitariste.

Pour le cas de la division d'une ressource divisible entre deux agents, l'algorithme *divide-and-choose* consiste à laisser un agent diviser la ressource en deux, puis à laisser l'autre agent choisir sa part. Cet algorithme est garanti de générer une allocation sans-envie et proportionnelle, mais pas nécessairement efficace. Les travaux de Plaut et Roughgarden [2018] tentent d'adapter cet algorithme au cas de ressources indivisibles, mais au prix d'une complexité exponentielle.

Pour le cas de l'allocation d'un ensemble de ressource divisible entre deux agents, il existe l'algorithme *adjusted winner*, proposé par Brams et Taylor [1996]. Cette méthode consiste à trier les objets selon le ratio entre les utilités qu'ils apportent à chaque agent, puis à allouer tous les biens pour lesquels ce ratio est plus grand que 1 au premier agent, et enfin, à équilibrer les lots des deux agents en transférant des objets, puis une fraction d'objet, vers le moins bien doté. Cette méthode garantit une solution Pareto-optimale et sans envie, elle permet aussi de ne diviser qu'un seul objet. La distance d'une solution retournée par cet algorithme à l'utilité optimale que l'on pourrait obtenir dans une telle instance a été étudiée par Bei *et al.* [2021].

Exemple 1.2.3: Algorithme d'adjusted winner

Considérons une instance du problème d'allocation de ressource divisibles constituée d'un ensemble d'agents $\mathcal{N} = \{a_1, a_2\}$ et d'un ensemble de ressources $\mathcal{M} = \{h_1, h_2, h_3, h_4, h_5, h_6\}$, et posons les utilités suivant le tableau ci-dessous :

	h_1	h_2	h_3	h_4	h_5	h_6
a_1	5	5	8	1	6	2
a_2	2	1	4	3	2	4

On ordonne les objets h par la valeur de leur rapport $\frac{u_{a_1}(h)}{u_{a_2}(h)}$ par ordre décroissant. Nous obtenons $h_2, h_5, h_1, h_3, h_6, h_4$ (car $5 > 3 > \frac{5}{2} > 2 > \frac{1}{2} > \frac{1}{3}$), et la limite du ratio de 1 est franchie entre h_3 et h_6 .

Nous allouons donc dans un premier temps h_2, h_5, h_1 et h_3 à a_1 et h_6 et h_4 à a_2 . L'utilité de a_1 est alors de 24 tandis que celle de a_2 est de 7, nous allons donc transférer des objets de a_1 à a_2 en commençant par ceux dont le ratio est le plus faible.

Nous transférons l'objet h_3 de a_1 à a_2 , les nouvelles valeurs d'utilités sont 16 pour a_1 et 10 pour a_2 . À ce stade, transférer l'objet h_1 de a_1 à a_2 mènerait l'utilité de a_2 à dépasser l'utilité de a_1 (12 contre 11). Il nous faut donc diviser l'objet entre les deux agents.

L'allocation finale A est donc telle que a_1 reçoit h_2, h_5 et $\frac{1}{7}$ de h_1 tandis que a_2 reçoit h_3, h_4, h_6 et $\frac{6}{7}$ de h_1 (chaque agent obtenant une utilité de $\frac{78}{7}$)

Enfin, un dernier mécanisme pour générer des allocations est celui de *picking sequence* ou lotterie. Le concept est simple, les agents sont ordonnés et vont chacun leur tour choisir un objet. Ce mécanisme, formalisé par Kohler et Chandrasekaran [1971], a été très étudié dans le cadre de la théorie des jeux, mais il est pertinent dans notre cadre car il est parfois utilisé dans les problèmes réels dont s'inspirent les modèles (par exemple pour les travaux de Benabbou *et al.* [2018]). Les allocations que cette méthode peut produire seront analysées plus en détail dans le chapitre 2.

Une autre méthode sur laquelle nous reviendrons plus tard est celle des échanges améliorants, utilisée par Damamme *et al.* [2015] dans le cadre d'un problème de *house allocation*. Cette méthode consiste à permettre aux agents d'échanger leur ressource entre eux, sous réserve que chaque échange améliore le lot de chaque agent impliqué.

1.3 Diversité

Nous allons maintenant nous concentrer sur la notion de diversité, qui est une notion plus rarement considérée dans la littérature que l'efficacité et l'équité. Les problèmes où la notion de diversité est importante sont principalement ceux qui se rapportent à des situations réelles où des autorités souhaitent s'assurer d'une certaine mixité eth-

nique (problème de l'immobilier de Singapour, décrit par Benabbou *et al.* [2018]) ou socio-économique (problème des écoles de Chicago, décrit par U.S. Department of Education, Office of Elementary and Secondary Education [2017]). Dans un problème considérant la notion de diversité, les agents seront répartis en types, et l'on cherchera à satisfaire une certaine mesure de diversité dans la répartition des types de l'allocation finale. Dans certains problèmes (tels que ceux considérés par Echenique et Yenmez [2015] et Benabbou *et al.* [2018]), chaque agent appartient à un unique type, l'ensemble des types forme donc une partition de l'ensemble des agents. Dans d'autres modèles (tels que ceux considérés par Aziz et Sun [2021]), les agents seront autorisés à faire partie de plusieurs types en même temps, le modèle contiendra donc une fonction de type T qui à un agent i renverra l'ensemble $T(i)$ de ses types. Les ressources peuvent elles aussi être réparties en groupes, mais les modèles existants dans ce cas utilisent tous une partition sur l'ensemble des ressources : Benabbou *et al.* [2018] partitionnent cet ensemble en un certain nombre de blocs, tandis que dans les modèles de choix d'écoles (par exemple Aziz et Sun [2021]) les contrats sont chacun liés à une école unique.

Pour s'assurer d'une certaine diversité dans les allocations finales, les modèles prenant en considération la diversité imposent des contraintes sur les allocations autorisées. Echenique et Yenmez [2015] présentent deux contraintes particulières : le **point idéal**, où une certaine proportion d'agents de chaque type sera sélectionné, et les **fonctions concaves de Schur**, une relaxation de la méthode précédente qui essaie de maximiser une fonction concave. Cependant, la grande majorité de la littérature se concentre sur deux méthodes particulières : les **quotas maximums** et les **places réservées**. Les **quotas maximums**, tels qu'utilisés par Benabbou *et al.* [2018], reposent sur un concept simple : chaque type d'agents a un quota maximum pour un bloc d'objet donné, et une allocation ne peut pas avoir une plus grande proportion d'agents de ce type dans ce bloc.

Exemple 1.3.1: Quotas maximums

Considérons une instance avec un ensemble d'agents \mathcal{N} divisé en deux types $T_1 = \{a_1, a_2\}$ et $T_2 = \{a_3, a_4\}$ et un ensemble d'objets \mathcal{M} divisé en deux blocs $B_1 = \{h_1, h_2\}$ et $B_2 = \{h_3, h_4\}$. On suppose les quotas maximums établis de la manière suivante :

	B_1	B_2
T_1	2	1
T_2	1	2

Dans cet exemple, les quotas autorisent les deux agents de T_1 à être dans B_1 (c'est-à-dire qu'ils peuvent obtenir h_1 et h_2), mais ils n'autorisent qu'un seul de ces agents à être dans B_2 (donc si l'un d'eux obtient h_3 ou h_4 , l'autre doit obligatoirement obtenir h_1 ou h_2).

Les **places réservées**, de leur côté, reposent sur le principe inverse, et désignent des

quotas minimum : chaque école doit réserver un certain nombre de place pour chaque type.

Exemple 1.3.2: Places réservées

Considérons une instance avec un ensemble d'agents \mathcal{N} divisé en deux types $T_1 = \{a_1, a_2\}$ et $T_2 = \{a_3, a_4\}$ et un ensemble d'objets \mathcal{M} divisé en deux blocs $B_1 = \{h_1, h_2\}$ et $B_2 = \{h_3, h_4\}$. On suppose les quotas minimums (places réservées) établis de la manière suivante :

	B_1	B_2
T_1	1	0
T_2	0	1

Dans cet exemple, B_1 a une place réservée pour T_1 , l'un de ses objets (h_1 ou h_2) doit donc être alloué à un agent de T_1 (a_1 ou a_2).

Puisqu'il est parfaitement possible pour ces deux critères d'être appliqués en même temps, cette deuxième méthode est souvent utilisée en même temps que la première, par exemple par Aziz *et al.* [2019b] ou Sankar *et al.* [2022]. Notons également que lorsque le modèle autorise les agents à appartenir à plusieurs types en même temps, il est nécessaire de choisir si un agent compte pour un seul quota ou plusieurs en même temps (Ehlers *et al.* [2014] discutent des différences de modèle entre les deux possibilités).

Si la majorité des travaux traitant de diversité utilisent les quotas maximums et/ou minimum comme des contraintes strictes, d'autres se contentent de les utiliser pour orienter le résultat, en pénalisant les allocations qui ne les satisfont pas. Ce concept de *soft bounds*, introduit par Ehlers *et al.* [2014], consiste à autoriser les agents à enfreindre les quotas, mais imposent des contraintes particulières quand les quotas sont enfreints. Les différents travaux sur le problème du choix des écoles utilisent plusieurs concepts spécifiques en relation avec ces quotas. Ehlers *et al.* [2014] introduisent le concept d'*envie justifiée*, qui adapte la définition de paire instable des problèmes de two-sided matching en lui interdisant d'enfreindre les quotas, tandis que Kurata *et al.* [2015] permettent aux agents de *demandeur une place vide* ("claiming an empty seat") si ils préfèrent une école pour laquelle leur quotas minimum n'a pas été atteint (ou si il existe réellement une place vide). Ces nouvelles notions servent alors de critères de qualité pour la solution : plutôt que de chercher une allocation stable respectant strictement les quotas, on cherche une allocation sans envie justifiée où aucun agent ne demande de place vide.

En raison de la grande variété de modèles, les mécanismes utilisés pour générer des allocations satisfaisant ces contraintes de diversité diffèrent. Pour le contexte du problème de house allocation, Benabbou *et al.* [2018] proposent une approximation en réduisant le problème à celui du *bounded color matching*, qui possède un algorithme d'approximation polynomial. Pour le contexte du problème de choix des écoles, et

plus précisément le cas où chaque agent ne compte que pour un seul quota, Kurata *et al.* [2015] proposent un algorithme appelé *Deferred Acceptance for Overlapping Types* capable de générer des allocations stables. Un algorithme pour le cas où un agent compte pour chacun de ses types est proposé par Aziz *et al.* [2020], appelé *Generalized Deferred Acceptance for Type Combination*.

1.3.1 Conclusion

Comme nous l'avons vu dans l'introduction, les modèles considérés dans ce manuscrit combinent les spécificités des problèmes de *house allocation* avec des types d'agents. Il existe de nombreux travaux dans la littérature qui considèrent des problèmes d'immobilier en conjonction avec des types d'agents, mais peu de ces travaux appartiennent au domaine du partage équitable. Les travaux sur les jeux de Schelling en particulier considèrent un problème qui est proche du nôtre, mais avec une approche et des objectifs très différents. Par conséquent, nous essaierons dans un premier temps d'adapter des notions utilisés dans ces modèles à notre problème : en l'occurrence, l'homophilie (c'est-à-dire, la volonté des agents de se rassembler entre agents de même type) et la dynamique d'échanges entre agent.

En ce qui concerne l'équité entre types dans les allocations, les modèles utilisés dans ce manuscrit se démarquent de leurs équivalents dans la littérature sur le partage équitable en étant une combinaison de nombreux cas particuliers. En effet, si il existe déjà des travaux sur l'équité entre types d'agents, la très grande majorité se restreint à des fonctions d'utilités additives. Or, nous nous plaçons dans un modèle de *house allocation*, ce qui veut dire que chaque agent ne peut utiliser qu'un seul objet à la fois. Par conséquent, les fonctions d'utilité des types, que nous définirons comme les sommes des utilités des agents qui les composent, ne seront pas additives (par exemple, si un type reçoit trop d'objets, ces derniers auront une utilité marginale de zéro). Il existe bien sûr des résultats dans la littérature concernant instances avec des fonctions d'utilité quelconques ou sous-modulaires, en revanche ils concernent généralement des modèles plus simples et plus classiques. Il va donc nous être nécessaire d'adapter ou de redéfinir les notions que l'on souhaite utiliser. En l'occurrence, la notion que nous avons choisi de considérer est la notion d'envie, qui a déjà été étudié dans le cadre de types d'agents ainsi que dans le cadre de fonctions d'utilité non-additives, mais jamais les deux en même temps.

Chapitre 2

Stabilité sous contraintes de diversité

2.1 Introduction

Le modèle de l'immobilier de Singapour, décrit par Benabbou *et al.* [2018], est un modèle utilisé pour étudier la répartition des logements publics de Singapour entre les demandeurs de logements par un organisme gouvernemental appelé "Housing and Development Board". Retrascriptant les objectifs de l'organisme en termes de diversité ethnique, ce modèle est une variante du problème de *house allocation* où les agents sont répartis en types (représentant des groupes ethniques) et les objets en blocs (représentant les quartiers où sont présents les logements), et où des quotas maximum sont imposés entre les types et les blocs. Les problèmes de *house allocation* ne permettant qu'un seul objet alloué à chaque agent, certains calculs deviennent plus difficiles, et si la maximisation de l'utilité des agents y reste calculable en temps polynomial, Benabbou *et al.* [2018] ont montré que l'ajout des quotas maximums suffisaient à rendre ce calcul NP-complet. Si le modèle a été initialement prévu pour étudier l'immobilier de Singapour, il permet aussi d'étudier des problèmes tels que l'allocation des élèves aux écoles publiques de Chicago : les élèves sont groupés par catégorie socio-économique, on souhaite leur attribuer des places en école, en s'assurant d'une certaine mixité socio-économique dans chaque école. Un autre problème pouvant être représenté par ce modèle est celui de l'admission post-bac : les élèves sont boursiers ou non-boursiers, et chaque université doit réserver un certain nombre de place pour les élèves boursiers.

Un modèle très différent étudiant des problématiques proches est celui des jeux de Schelling (Schelling [1969]), déjà mentionné dans le chapitre 1. Ces modèles ne s'intéressent pas à distribuer des logements mais plutôt à l'évolution dans le temps de leurs occupants, en s'intéressant tout particulièrement aux comportements des habitants en fonction de leurs voisins actuels ou projetés. Ils accomplissent en plaçant des agents sur un graphe, puis en permettant aux agents mécontents de se déplacer (un exemple de jeu de Schelling est donné dans l'exemple 1.1.5). Ce déplacement, selon les modèles, constituera en un échange de position entre deux agents mécontents, ou

en un déplacement sur un sommet inoccupé. Le modèle est alors utilisé pour étudier l'évolution de la répartition des agents. Si le modèle d'origine se contente de vérifier si les agents sont satisfaits ou non par cette proportion, des travaux ultérieurs sur des variantes du modèle (Elkind *et al.* [2019] et Kanellopoulos *et al.* [2020]) vont représenter le degré de satisfactions des agents par une fonction d'utilité, qui à une allocation des agents aux sommets retourne la proportion de voisin de même type. Chauhan *et al.* [2018], de leur côté, cherchent à ajouter aux agents des préférences sur les positions : chaque agent a une position cible, son coût dans une allocation est le couple de sa distance à sa cible et de la différence entre ses proportions de voisin souhaitée et réelle. Ainsi, chaque agent cherche à concilier deux objectifs : d'une part il souhaite se placer sur un sommet qui est lui-même efficace, d'autre part il souhaite se coordonner avec les agents de son type. Ces comportements peuvent être rapprochés de certains travaux sur des problèmes de *house allocation*, dans le modèle de Elkind *et al.* [2020], on souhaite allouer des logements représentés par les sommets d'un graphe, deux logements étant voisins s'ils sont reliés par une arête. Les agents ont des amis, et souhaitent à la fois obtenir un logement qui leur plaise, mais aussi se placer proche de leurs amis. Leurs préférences sont alors représentées par des fonctions d'utilité composées de deux parties, une qui va dépendre de la qualité du logement qui leur a été alloué, et une qui va dépendre de la présence éventuelle d'amis dans le voisinage. Cependant, l'article note qu'il est alors difficile de générer des allocations maximisant ces utilités.

L'objectif des travaux que nous allons présenter maintenant va être d'introduire les notions présentes dans les modèles modifiés des jeux de Schelling au problème de l'immobilier de Singapour. Puisque Elkind *et al.* [2020] ont montré que faire dépendre les utilités des agents des comportements d'autres agents rend le calcul d'une allocation optimale difficile, nous nous intéresserons plutôt à reproduire les dynamiques régissant les mouvements des agents dans les jeux de Schelling, pour lesquels des études de stabilité et d'efficacité ont été faites (par exemple par Agarwal *et al.* [2020]).

Nous commencerons par introduire notre modèle ainsi que les résultats les plus basiques qui s'y rapportent. Nous définirons ensuite une dynamique basée sur les comportements individuels des agents, et commenceront notre étude théorique. Nous étudierons ensuite deux mécanismes, le mécanisme de loterie d'abord, utilisé en situation réelle, puis le mécanisme d'échanges bilatéraux, basé sur notre dynamique. Nous finirons par discuter des résultats de plusieurs séries d'expérimentations que nous avons effectuées avec ces mécanismes.

2.2 Modèle

2.2.1 Instances et Allocations

Nous allons considérer une version légèrement modifiée du modèle de Benabbou *et al.* [2018].

Définition 2.2.1: Instance du problème d'appariement sous contraintes de diversité

Une instance I du modèle contient les éléments suivant :

- un ensemble \mathcal{N} d'agents $\mathcal{N} = \{a_1, a_2, a_3, \dots, a_n\}$ partitionnés en k types $\mathcal{T} = \{T_1, T_2, \dots, T_k\}$.
- un ensemble \mathcal{M} d'objets $\mathcal{M} = \{h_1, h_2, h_3, \dots, h_m\}$ partitionnés en l blocs $\mathcal{B} = \{B_1, B_2, \dots, B_l\}$.
- pour chaque agent $a_i \in \mathcal{N}$, une fonction d'utilité de position $u_i^I : \mathcal{M} \rightarrow [0, 1]$ représentant les préférences de l'agent a_i sur les objets.
- une matrice de quotas de diversité $\Lambda = \{\lambda_{p,q}\}_{p \in \llbracket 1;k \rrbracket, q \in \llbracket 1;l \rrbracket} \in \mathbb{N}^{k \times l}$. Le quota $\lambda_{p,q}$ est le nombre maximum d'agents de type T_p pouvant être alloués à des objets du bloc B_q .

On impose les contraintes suivantes sur les quotas :

- $\forall p \in \llbracket 1;k \rrbracket, q \in \llbracket 1;l \rrbracket, \lambda_{p,q} \leq |B_q|$
- $\forall q \in \llbracket 1;l \rrbracket, \sum_{p \in \llbracket 1;k \rrbracket} \lambda_{p,q} \geq |B_q|$

La première contrainte assure qu'un quota sur un bloc ne soit pas supérieur au nombre de biens dans un bloc. Imposer cette contrainte ne restreint pas le modèle, mais permet de clarifier les calculs. La deuxième contrainte impose que la somme des quotas d'un bloc ne soit pas inférieure au nombre d'objets. Contrairement à la première, cette contrainte empêche certaines configurations. Cependant, le cas où tous les biens ne peuvent pas être alloués n'entre pas dans le cadre de cette étude.

Exemple 2.2.1: Instance

Nous définissons l'instance suivante :

T_1 : 

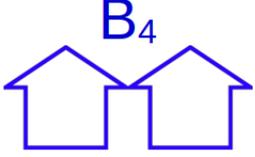
T_2 : 

T_3 : 

B_1 : 

B_3 : 

B_2 : 

B_4 : 

Λ :					
		3	1	2	1
		1	1	1	1
		2	1	1	1

U :					
		0,4	0,7	0,5	0,2
		0,8	0,1	0,9	0,4
		0,6	0,3	0,1	0,7

Dans cette instance, nous avons 3 types T_1 , T_2 et T_3 de tailles respectives 4, 2 et 3, et 4 blocs B_1 , B_2 , B_3 et B_4 , de tailles respectives 3, 1, 2 et 2.

Le premier tableau donne la matrice Λ des quotas (par exemple, $\lambda_{2,3} = 1$) tandis que le deuxième tableau (U) donne les utilités de position. Nous supposons ici que les utilités de position sont identiques entre les agents d'un même type et indifférentes entre les objets d'un même bloc (ainsi, un objet de B_3 apporte une utilité de 0,5 à un agent de T_1). Nous pouvons donc définir directement ces utilités entre les types et les blocs plutôt qu'entre les agents et les objets.

Définition 2.2.2: Allocation

Une **allocation** est une fonction $A : \mathcal{N} \rightarrow \mathcal{M} \cup \{\emptyset\}$ qui à chaque agent associe un unique objet (ou l'ensemble vide, si l'agent ne reçoit aucun objet).

40

Définition 2.2.3: Allocation valide

On dit qu'une allocation est **valide** si elle vérifie les conditions suivantes :

- $\forall i \in \mathcal{N}, j \in \mathcal{N}, A(i) \neq A(j)$ ou $A(i) = A(j) = \emptyset$ (aucun objet n'est partagé)
- $\bigcup_{i \in \mathcal{N}} A(i) = \mathcal{M}$ (tous les objets sont alloués à des agents)
- $\forall p \in \llbracket 1; k \rrbracket, q \in \llbracket 1; l \rrbracket, |\{i \in T_p | A(i) \in B_q\}| \leq \lambda_{p,q}$ (les contraintes de diversité sont respectées)

La deuxième condition nécessite que $|\mathcal{N}| \geq |\mathcal{M}|$ pour qu'une allocation valide existe, nous imposerons donc cette contrainte dans la suite.

Notons qu'il est possible de déterminer si une instance donnée admet une allocation valide en temps polynomial. Cela a été montré par Benabbou *et al.* [2018] en utilisant une réduction à un problème de flots.

2.2.2 Utilité des agents, utilité globale

Chaque agent possède une fonction d'utilité représentant ses préférences. Ces préférences ne dépendent habituellement que de l'objet reçu, mais suivant les exemples de Chauhan *et al.* [2018] et Elkind *et al.* [2020], nous souhaitons également représenter le fait que l'agent cherche à être proche des autres membres de son type. Cette fonction sera donc composé de deux parties : une partie qui va dépendre de l'objet alloué à l'agent, et une partie qui va dépendre des voisins de l'agent, c'est-à-dire des agents alloués à des objets du même bloc. On définit :

Définition 2.2.4: Utilité de position

L'utilité de position d'un agent $i \in \mathcal{N}$ est une fonction $u_i^I : \mathcal{M} \rightarrow [0, 1]$ telle que $u_i^I(h)$ indique l'utilité de l'objet h pour l'agent i . On étend cette fonction aux allocations en posant $u_i^I(A) = u_i^I(A(i))$ (l'utilité de position d'un agent i dans l'allocation A est égale à l'utilité de position que i obtient de l'objet qui lui est alloué dans l'allocation A)

Cette partie est appelé utilité "de position" car elle est liée à la position où l'agent va être alloué, mais elle peut représenter bien plus que l'intérêt que l'agent porte à la localisation des objets: il s'agit, de fait, de l'utilité classique, et représente donc les préférences de l'agent pour les caractéristiques diverses des objets (taille, montant du loyer, agencement,...).

Définition 2.2.5: Utilité de voisinage

L'utilité de voisinage u_i^N d'un agent i est l'utilité qu'un agent obtient de la constitution de son voisinage. Nous la définissons comme égale à la proportion d'agents de même type alloués à des objets du même bloc. En d'autres termes, pour un agent $i \in T_p$ tel que $A(i) \in B_q$:

$$u_i^N(A) = \frac{|\{j \in T_p | A(j) \in B_q\}|}{|B_q|}$$

Un détail important à noter est que l'agent se compte lui-même comme un voisin du même type. La logique derrière ce choix est apparente lorsque l'on considère un agent seul : si il ne peut pas obtenir de voisin du même type, il préférera être dans un bloc de petite taille, où il représentera une plus grosse proportion des occupants. En particulier, il préférera être seul plutôt qu'en présence d'un grand nombre d'agents d'autres types.

Définition 2.2.6: Utilité des agents

L'utilité $u_i(A)$ de l'agent i dans l'allocation A sera alors :

$$u_i(A) = u_i^I(A(i)) + \varphi u_i^N(A)$$

Où φ est un facteur déterminant l'importance relative de l'utilité apportée par le voisinage appelé le **compromis d'utilité**.

Nous considérerons dans la suite que le compromis d'utilité est identique pour tous les agents, même si nous mentionnerons parfois le cas où chaque agent i peut avoir un compromis d'utilité φ_i différent.

Cette utilité de voisinage est inspirée des travaux sur les jeux de Schelling (tels que Elkind *et al.* [2019] et Kanellopoulos *et al.* [2020]), cependant, là où ces derniers utilisaient un graphe pour induire une topologie sur les relation de voisinage, nous nous contenterons de considérer que tous les agents alloués aux objets d'un même bloc sont voisins les uns des autres. Notre manière de calculer l'utilité peut donc être comparée à un jeu de Schelling sur un graphe constitué de cliques séparées.

Certains modèles des jeux de Schelling (comme Elkind *et al.* [2019]), plutôt que d'agréger les deux utilités comme nous le faisons, considèrent deux catégories d'agents : les agents têtus, qui ne considèrent que le sommet sur lequel ils se trouvent pour calculer leur utilité, et les agents stratégiques, dont l'utilité ne prend en compte que les voisins. À certains points, nous considérerons des agents aux comportements similaires, nous définissons donc ici deux termes référant à ces cas particuliers :

- les agents sont **item-focused** ("centrés sur les objets") lorsqu'ils ne considèrent que l'objet qu'ils obtiennent dans le calcul de leur utilité (c'est-à-dire $\varphi = 0$).

L'instance est alors équivalente à une instance du problème de l'immobilier de Singapour classique tel que décrit par Benabbou *et al.* [2018].

- les agents sont **neighbour-focused** ("centrés sur les voisins") lorsqu'ils ne considèrent que leurs voisins dans le calcul de leur utilité (c'est-à-dire $\varphi \neq 0$ and $\forall i \in \mathcal{N}, h \in \mathcal{M}, u_i(h) = 0$).

Lorsque nous souhaiterons estimer la qualité de l'état global du système dans une allocation A , nous utiliserons le bien-être social utilitariste $sw(A)$, c'est-à-dire la somme des utilités des agents :

$$sw(A) = \sum_{i \in \mathcal{N}} u_i(A)$$

Nous appellerons parfois cette valeur l'utilité globale.

2.2.3 Amplitude maximum de l'utilité

Nous savons que calculer une allocation maximisant cette utilité sera NP-difficile dans le cas général. En effet, pour le cas particulier où tous les agents sont item-focused, nous pouvons utiliser le résultat de NP-complétude apporté par Benabbou *et al.* [2018]. Par conséquent, les mécanismes dont nous discuterons dans la suite pourront au mieux approcher cette valeur. Afin de pouvoir mettre en place des comparaisons pertinentes, nous commençons par définir l'amplitude maximum de cette utilité globale sur l'ensemble des allocations valides.

Lemme 2.2.1

Le quotient maximum entre le bien-être social de deux allocations est majoré par $\frac{1+\varphi}{\varphi} \cdot k$, où k est le nombre de types, quand $\varphi \in]0, 1]$ (et est non borné si $\varphi = 0$). Plus spécifiquement, si A^\top est l'allocation de bien-être social maximum et A^\perp est l'allocation de bien-être social minimum, alors :

$$\frac{sw(A^\top)}{sw(A^\perp)} \leq \frac{1+\varphi}{\varphi} \cdot k$$

où k est le nombre de types.

De plus, il existe des instances qui atteignent cette borne.

Preuve. L'existence de la borne supérieure se montre en bornant les utilités qu'une allocation peut atteindre. On sait que, si la valeur du compromis d'utilité de l'instance est $\varphi \in [0, 1]$, alors la valeur d'utilité maximum d'une allocation pour un agent alloué à un bien vaut $1 + \varphi$. L'utilité globale maximum vaut donc $|\mathcal{M}|(1 + \varphi)$

Il nous faut maintenant montrer que l'utilité globale minimum d'une allocation A est bornée par $|\mathcal{M}| \frac{\varphi}{k}$. Il est évident que l'utilité de position minimum d'un agent est de zéro, et qu'il est tout à fait envisageable, sur un problème approprié, que tous les

agents aient simultanément une utilité de zéro. Il nous reste donc à minimiser l'utilité de voisinage. Pour cela, notons tout d'abord que dans un bloc $B_q \in \mathcal{B}$, l'utilité de voisinage d'un agent i de type $T_p \in \mathcal{T}$ dans l'allocation A telle que $A(i) \in B_q$ vaut :

$$u_i^N(A) = \frac{|\{j \in T_p | A(j) \in B_q\}|}{|B_q|}$$

Par conséquent, l'utilité de voisinage combinée des agents de type T_p alloués à des objets du bloc B_q vaut :

$$\sum_{i \in T_p, A(i) \in B_q} u_i^N(A) = \frac{|\{j \in T_p | A(j) \in B_q\}|^2}{|B_q|}$$

Dans une allocation où l'utilité de position serait égale à zéro pour tous les agents, le bloc B_q contribuerait donc à l'utilité globale pour une valeur de :

$$\sum_{p=1}^k \sum_{i \in T_p, A(i) \in B_q} u_i^N(A) = \sum_{p=1}^k \frac{|\{j \in T_p | A(j) \in B_q\}|^2}{|B_q|}$$

Nous pouvons calculer la valeur minimum de cette utilité en renommant $|\{j \in T_p, A(j) \in B_q\}|$ en $x_{p,q}$ et en posant le programme linéaire suivant :

$$\begin{aligned} \min \quad & \frac{1}{|B_q|} \sum_{p=1}^k x_{p,q}^2 \\ \text{s.t.} \quad & \sum_{p=1}^k x_{p,q} = |B_q| \\ & x_{p,q} \in \mathbb{R}_+, \forall p \in [k] \end{aligned}$$

Notons que nous ne considérons pas ici les contraintes de diversité, qui réduiraient l'ensemble des valeurs possibles. Il est possible de les ignorer puisqu'elles n'impactent pas la borne inférieure, qui doit minorer même les problèmes aux quotas très permissifs.

La solution optimale x_q^* du programme linéaire est telle que :

$$\exists \lambda \in \mathbb{R}, \nabla f(x^*) + \lambda \nabla g(x^*) = 0$$

où $f(x) = \frac{1}{|B_q|} \sum_{p=1}^k x_{p,q}^2$ and $g(x) = \left(\sum_{p=1}^k x_{p,q} \right) - |B_q|$.

Nous pouvons donc en déduire que x_q^* et λ sont tels que :

$$\frac{2}{|B_q|} \begin{pmatrix} x_1^* \\ \vdots \\ x_k^* \end{pmatrix} + \lambda \begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix} = 0$$

et par conséquent $x_{p,q}^* = -\frac{\lambda|B_q|}{2}$ pour tout $p \in [k]$.

Puisque nous avons aussi $\sum_{p=1}^k x_k^* = |B_q|$, nous en déduisons que $\lambda = -\frac{2}{k}$ et que $x_p^* = \frac{|B_q|}{k}$ pour tout $p \in [k]$. Nous obtenons donc la borne inférieure suivante sur les utilités de voisinages des agents alloués à des objets de B_q :

$$u_i^N(A) \geq \frac{|B_q|}{k} \times \frac{1}{|B_q|}$$

Nous pouvons utiliser ce résultat pour borner la contribution du bloc B_q à l'utilité globale

$$\sum_{\substack{i \in \mathcal{N} \\ A(i) \in B_q}} u_i^N(A) \geq f(x^*) = \frac{1}{|B_q|} \sum_{p=1}^k \left(\frac{|B_q|}{k} \right)^2 = \frac{|B_q|}{k}$$

Pour toute allocation valide A . Cela implique une borne sur le bien-être social pour toute allocation valide A :

$$sw(A) \geq \sum_{i \in \mathcal{N}} \varphi \cdot u_i^N(A) = \varphi \sum_{B_q \in \mathcal{B}} \sum_{\substack{i \in \mathcal{N} \\ A(i) \in B_q}} u_i^N(A) \geq \varphi \sum_{B_q \in \mathcal{B}} \frac{|B_q|}{k} = \frac{\varphi \cdot |\mathcal{M}|}{k}$$

Le plus grand quotient entre les utilités globale de deux allocations est donc borné par :

$$\frac{sw(A^\top)}{sw(A^\perp)} \leq \frac{|\mathcal{M}|(1+\varphi)}{\frac{\varphi \cdot |\mathcal{M}|}{k}} = \frac{1+\varphi}{\varphi} \cdot k$$

Il nous reste maintenant à montrer que ce quotient est atteint. Pour cela, nous allons simplement créer une instance qui les atteint.

Considérons un paramètre $k \in \mathbb{N}$. Nous définissons l'instance I_k à partir de k de la manière suivante : l'instance I_k est constituée de k^2 agents répartis en k types de k agents chacun, et de k^2 objets répartis en k blocs de k objets chacun. Les quotas de diversité sont tous placés à k (et n'apportent donc aucune contrainte).

Nous étiquetons les agents et les objets de la manière suivante : pour $i \in [[1, k]]$ et $p \in [[1, k]]$, l'agent $a_{p,i}$ est l'agent numéro i du type T_p et pour $j \in [[1, k]]$ et $q \in [[1, k]]$, l'objet $h_{q,j}$ est l'objet numéro j du bloc B_q . Nous définissons alors les fonctions d'utilités des agents de la manière suivante :

$$u_{p,i}^I(h_{q,j}) = \begin{cases} 1 & \text{si } p = q \text{ et } i = j, \\ 0 & \text{sinon} \end{cases}$$

Nous considérons deux allocations :

- l'allocation A_1 telle que $\forall i \in [[1, k]], p \in [[1, k]], A(a_{p,i}) = h_{p,i}$

- l'allocation A_2 où chaque bloc contient un agent de chaque type (c'est-à-dire que si pour un type T_p , les objets $A(a_{p,i_1})$ et $A(a_{p,i_2})$ appartiennent au même bloc, alors $i_1 = i_2$), et où aucun agent n'est alloué à l'objet qui lui confère une valeur de 1 (c'est-à-dire que si $A(a_{p,i}) = h_{p,j}$ alors $i \neq j$).

Dans l'allocation A_1 , chaque agent a une utilité de $1 + \varphi$, puisqu'il a obtenu un objet d'utilité 1 et que tous ses voisins sont de son type. Dans l'allocation A_2 , chaque agent a une utilité de $\frac{\varphi}{k}$, puisqu'il a obtenu un objet d'utilité 0 et qu'aucun de ses voisins n'est de son type. Par conséquent, le quotient entre les utilités globales de ces deux allocations vaut :

$$\frac{sw(A_1)}{sw(A_2)} = \frac{|\mathcal{M}|(1 + \varphi)}{\frac{|\mathcal{M}|\varphi}{k}} = \frac{1 + \varphi}{\varphi} \cdot k$$

□

2.3 Prix de la stabilité

Une considération de première importance pour ces problèmes d'allocation de ressources est la question de la **stabilité** de l'allocation, que nous définissons ici comme la propension des agents à vouloir rester dans l'allocation actuelle. Nous dirons ici qu'une allocation est **stable** si elle ne contient pas de couple d'agents pouvant augmenter leurs utilités en échangeant leurs objets. Une allocation stable peut donc avoir plus de légitimité auprès des agents, qui n'ont pas immédiatement intérêt à en dévier. Elle sera également plus facile à mettre en place par un organisme gouvernementale, puisque les personnes concernés auront plus de mal à justifier d'éventuelles réclamations. Plus formellement, nous posons les définitions suivantes :

Définition 2.3.1: Échange mutuellement améliorant

Un échange d'objet entre deux agents i et j de \mathcal{N} est dit **mutuellement améliorant** dans l'allocation A si et seulement si on a $u_i(A(i)) < u_i(A_{i \leftrightarrow j}(i))$ et $u_j(A(j)) < u_j(A_{i \leftrightarrow j}(j))$ où $A_{i \leftrightarrow j}$ est l'allocation obtenue en échangeant l'allocation de i et j .

Un concept similaire avait déjà été étudié dans le cadre de problème de *house allocation* sans type d'agent par Damamme *et al.* [2015], qui s'inspiraient de travaux sur les *top trading cycles* (Shapley et Scarf [1974]). Ce concept se rapproche également des travaux de Chevalyere *et al.* [2010], qui utilisaient des échanges bilatéraux avec transfert d'argent, mais dans un cadre où plusieurs objets pouvaient être alloués à un même agent.

Définition 2.3.2: Échange valide

Dans une allocation A , un échange $i \leftrightarrow j$ entre les objets des agents i et j est **valide** si et seulement si l'allocation résultante $A_{i \leftrightarrow j}$ vérifie les contraintes de diversité.

Nous pouvons maintenant poser la définition précise de la stabilité :

Définition 2.3.3: Stabilité

Une allocation est dite **stable** si elle n'admet pas d'échange valide mutuellement améliorant.

De toute évidence, toutes les allocations ne seront pas stables. Nous pouvons alors nous demander dans quelle mesure la recherche d'une allocation stable peut coïncider avec la recherche d'une allocation optimale. Pour mesurer cela, nous définissons le prix de la stabilité.

Définition 2.3.4: Prix de la stabilité

Le prix de la stabilité PoS est défini comme le maximum sur toutes les instances I du quotient entre l'utilité d'une allocation optimale A_I^* , et l'utilité d'une allocation stable d'utilité maximum A_I^\top .

$$PoS = \sup_{I \in \mathcal{I}} \frac{sw(A_I^*)}{sw(A_I^\top)}$$

où \mathcal{I} est l'ensemble des instances possibles.

Proposition 2.3.1

Dans le cas où tous les agents sont item-focused, ainsi que dans le cas où tous les agents sont neighbour-focused, nous avons :

$$PoS = 1$$

Pour prouver ce résultat, nous aurons besoin du lemme suivant :

Lemme 2.3.1: Variation d'utilité lors d'un échange

Lors d'un échange valide entre deux agents a_1 et a_2 transformant l'allocation A en l'allocation A' , si on pose :

- $\Delta u_1^I = u_{a_1}^I(A') - u_{a_1}^I(A)$ la variation d'utilité de position de a_1
- $\Delta u_1^N = u_{a_1}^N(A') - u_{a_1}^N(A)$ la variation d'utilité de voisinage de a_1
- $\Delta u_2^I = u_{a_2}^I(A') - u_{a_2}^I(A)$ la variation d'utilité de position de a_2
- $\Delta u_2^N = u_{a_2}^N(A') - u_{a_2}^N(A)$ la variation d'utilité de voisinage de a_2
- $\Delta u^I = \Delta u_1^I + \Delta u_2^I$ la variation d'utilité de position des agents impliqués dans l'échange
- $\Delta u^N = \Delta u_1^N + \Delta u_2^N$ la variation d'utilité de voisinage des agents impliqués dans l'échange
- $\Delta U = sw(A') - sw(A)$ la variation d'utilité globale lors de l'échange

Alors on a :

$$\Delta U = \Delta u^I + 2.\Delta u^N$$

Preuve du lemme 2.3.1. Lors d'un échange entre deux agents, trois types de variation d'utilité vont se combiner dans la variation d'utilité globale ΔU :

- la variation d'utilité de position des agents impliqués dans l'échange Δu^I
- la variation d'utilité de voisinage des agents impliqués dans l'échange Δu^N
- la variation d'utilité de voisinage des agents qui ne sont pas impliqués dans l'échange, mais qui sont dans le type d'un des agents impliqués dans l'échange.

On regroupe cette dernière catégorie sous le terme Δu_{others}^N .

On a $\Delta u_{others}^N = \Delta U - \Delta u^I - \Delta u^N$. Il nous suffit donc de montrer que $\Delta u_{others}^N = \Delta u^N$.

On nomme T_{p_1} et T_{p_2} tels que $a_1 \in T_{p_1}$ et $a_2 \in T_{p_2}$. De même, on définit B_{q_1} et B_{q_2} tels que $A(a_1) \in B_{q_1}$ et $A(a_2) \in B_{q_2}$.

Nous nous restreindrons au cas $p_1 \neq p_2$ et $q_1 \neq q_2$. En effet, si $p_1 = p_2$ ou $q_1 = q_2$, alors $\Delta u_{others}^N = \Delta u^N = 0$ et le résultat est immédiat.

Nous noterons dans cette preuve les valeurs $A_{p,q}$ pour $p \in [[1, k]]$ et $q \in [[1, l]]$ telles que $A_{p,q} = |\{a \in T_p | A(a) \in B_q\}|$, pour clarifier les calculs d'utilité de voisinage.

On développe alors les calculs de Δu_{others}^N et Δu^N

$$\begin{aligned}
 \Delta u^N &= u_{a_1}^N(A') - u_{a_1}^N(A) + u_{a_2}^N(A') - u_{a_2}^N(A) \\
 &= \varphi \left(\frac{A_{p_1, q_2} + 1}{|B_{q_2}|} - \frac{A_{p_1, q_1}}{|B_{q_1}|} + \frac{A_{p_2, q_1} + 1}{|B_{q_1}|} - \frac{A_{p_2, q_2}}{|B_{q_2}|} \right) \\
 &= \varphi \left(\frac{A_{p_2, q_1} - A_{p_1, q_1} + 1}{|B_{q_1}|} + \frac{A_{p_1, q_2} - A_{p_2, q_2} + 1}{|B_{q_2}|} \right)
 \end{aligned}$$

$$\begin{aligned}
 \Delta u_{others}^N &= \sum_{\substack{i \in T_{p_1} \setminus \{a_1\} \\ A(i) \in B_{q_1} \cup B_{q_2}}} \left(u_i(A') - u_i(A) \right) + \sum_{\substack{i \in T_{p_2} \setminus \{a_2\} \\ A(i) \in B_{q_1} \cup B_{q_2}}} \left(u_i(A') - u_i(A) \right) \\
 &= (A_{p_1, q_1} - 1) \left(\frac{A_{p_1, q_1} - 1}{|B_{q_1}|} - \frac{A_{p_1, q_1}}{|B_{q_1}|} \right) \varphi + A_{p_1, q_2} \left(\frac{A_{p_1, q_2} + 1}{|B_{q_2}|} - \frac{A_{p_1, q_2}}{|B_{q_2}|} \right) \varphi \\
 &\quad + A_{p_2, q_1} \left(\frac{A_{p_2, q_1} + 1}{|B_{q_1}|} - \frac{A_{p_2, q_1}}{|B_{q_1}|} \right) \varphi + (A_{p_2, q_2} - 1) \left(\frac{A_{p_2, q_2} - 1}{|B_{q_2}|} - \frac{A_{p_2, q_2}}{|B_{q_2}|} \right) \varphi \\
 &= (A_{p_1, q_1} - 1) \frac{-1}{|B_{q_1}|} \varphi + A_{p_1, q_2} \frac{1}{|B_{q_2}|} \varphi + A_{p_2, q_1} \frac{1}{|B_{q_1}|} \varphi + (A_{p_2, q_2} - 1) \frac{-1}{|B_{q_2}|} \varphi \\
 &= \varphi \left(\frac{-(A_{p_1, q_1} - 1)}{|B_{q_1}|} + \frac{A_{p_1, q_2}}{|B_{q_2}|} + \frac{A_{p_2, q_1}}{|B_{q_1}|} + \frac{-(A_{p_2, q_2} - 1)}{|B_{q_2}|} \right) \\
 &= \varphi \left(\frac{A_{p_2, q_1} - A_{p_1, q_1} + 1}{|B_{q_1}|} + \frac{A_{p_1, q_2} - A_{p_2, q_2} + 1}{|B_{q_2}|} \right) \\
 &= \Delta u^N
 \end{aligned}$$

Par conséquent, $\Delta U = \Delta u^I + \Delta u_{others}^N + \Delta u^N = \Delta u^I + 2 \cdot \Delta u^N$ □

Nous pouvons maintenant revenir à la preuve de la proposition 2.3.1

Preuve de la proposition 2.3.1. Pour prouver cette égalité, il nous suffit de montrer que toute allocation optimale est un équilibre dans ces deux cas.

Le cas item-focused est immédiat : si deux agents peuvent augmenter leur utilité en échangeant leur biens, puisque l'utilité des autres agents ne sera pas impactée par cet échange, cela veut dire que l'on peut obtenir une allocation d'utilité strictement supérieure en échangeant les biens. Cela implique qu'une allocation optimale ne peut pas contenir d'échange valide améliorant, elle est donc stable.

Dans le cas neighbour-focused, le lemme 2.3.1 nous indique que la variation d'utilité lors d'un échange vaut $\Delta U = 2 \cdot \Delta u^N$. Or, pour un échange valide améliorant, on a nécessairement $\Delta u^N > 0$, puisque l'utilité des agents impliqués doit augmenter strictement. Par conséquent, si une allocation n'est pas stable, on peut augmenter son utilité par un échange, elle ne peut donc pas être optimale. Donc toute allocation optimale est stable. □

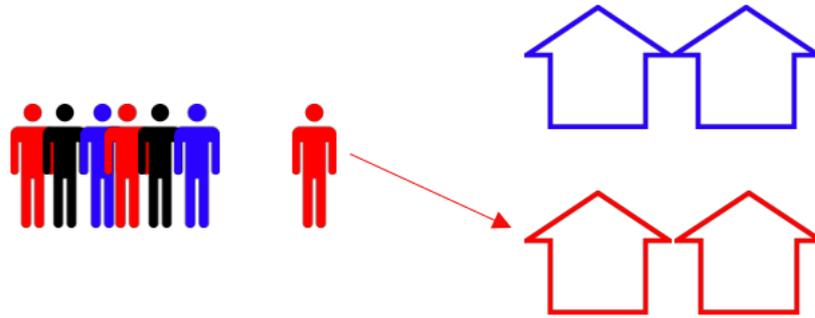


Figure 2.1: Mécanisme de la loterie

Proposition 2.3.2

Dans le cas général, nous avons :

$$1.6 \leq PoS \leq 2$$

Preuve : La preuve nécessite des éléments introduits plus tard lors d'une preuve de la section 2.5, nous la repoussons donc dans annexe B.

Après avoir considéré les propriétés théoriques de ce modèle, nous allons nous tourner vers l'étude des mécanismes de génération d'allocations. Nous allons considérer deux mécanismes principaux, le mécanisme de loterie, utilisée en situation réelle et dont l'étude a été entamée par Benabbou *et al.* [2018], et le mécanisme d'échanges rationnels, une continuation naturelle de notre étude sur la stabilité.

2.4 Mécanisme de loterie

Nous allons d'abord parler d'un mécanisme couramment utilisé pour calculer des allocations dans le modèle de Benabbou *et al.* [2018], le mécanisme de loterie. Ce mécanisme, aussi appelé *picking sequence*, a été le sujet de plusieurs études (Kohler et Chandrasekaran [1971], Bouveret et Lang [2011]), y compris son application aux problèmes de house allocation (voir Abdulkadiroğlu et Sönmez [1998]). Son fonctionnement est simple, en partant d'une allocation vide où aucun agent et aucun objet n'est alloué, on ordonne l'ensemble des agents selon un ordre aléatoire (d'où le terme de loterie). Puis, en suivant cet ordre, chaque agent choisit son objet préféré (en cas d'égalité, l'agent en choisit un au hasard parmi ses préférés) parmi les objets non alloués qui n'enfreignent pas les quotas de diversité. Si il ne peut prendre aucun objet sans enfreindre les quotas, l'agent passe son tour et ne se voit allouer aucun objet.

L'avantage de ce mécanisme, outre sa simplicité, est qu'il requiert très peu d'information sur les préférences des agents pour être effectué, puisqu'il suffit de demander à chaque

agent quel est son objet préféré parmi les objets libres qui n'enfreindraient pas les quotas.

Cependant, puisque les agents ne considèrent que leur propre utilité en choisissant leur objet, ils ne choisiront pas de manière à s'assurer qu'on puisse toujours allouer le reste des objets en respectant les quotas. Par conséquent, ce mécanisme ne retournera pas toujours une allocation valide.

Proposition 2.4.1

Le mécanisme de loterie ne retourne pas toujours une allocation valide, même si tous les agents sont neighbour-focused.

Preuve. Pour prouver cela, il suffit de montrer un exemple sur lequel ce mécanisme peut retourner une allocation non valide. Considérons une instance I avec deux types de deux agents chacun : $T_1 = \{a_1, a_2\}$ et $T_2 = \{a_3, a_4\}$, et deux blocs de deux objets chacun : $B_1 = \{h_1, h_2\}$ et $B_2 = \{h_3, h_4\}$. Nous posons les quotas suivant : $\lambda_{1,1} = \lambda_{2,1} = 1$ et $\lambda_{1,2} = \lambda_{2,2} = 2$, ce qui veut dire que le bloc B_1 ne peut contenir qu'un seul agent de chaque type alors que le bloc B_2 n'a pas de contrainte. Nous supposons que tous les agents utilisent la même fonction d'utilité de position u^I qui est telle que $u^I(h_1) = u^I(h_2) = 0$ et $u^I(h_3) = u^I(h_4) = 0.1$.

Si l'on applique le mécanisme de loterie sur cet exemple avec l'ordre $(1, 2, 3, 4)$, alors les deux premiers agents vont prendre les deux objets du bloc B_2 , ce qui force l'agent a_3 à prendre un objet de B_1 , laissant l'agent a_4 incapable de prendre le dernier objet sans enfreindre les quotas de diversité. Par conséquent, l'allocation résultante ne sera pas valide, puisque un agent et un objet ne seront pas alloués. □

Puisqu'il n'est pas garanti que le mécanisme retourne une allocation valide, nous pouvons nous demander s'il est possible de prédire quelles instances risquent de retourner une allocation non valide. Une condition suffisante pour que tous les ordres possibles retournent une allocation valide est que $\sum_{q=1}^l \lambda_{p,q} \leq |T_p|$ pour tout $p \in [k]$, c'est-à-dire que tous les types d'agents disposent d'une population suffisante pour remplir simultanément tous les quotas à leur disposition. (Par exemple, dans l'exemple précédent, si T_1 disposait d'un troisième agent, ce dernier pourrait être alloué au dernier bien restant.)

Nous allons maintenant nous intéresser à la qualité des allocations retournées et étudier la perte maximum dans le pire cas, définie comme la valeur maximum possible, sur toutes les instances, du rapport entre les utilités de l'allocation optimale et de la pire allocation qui puisse être retournée par le mécanisme. Rappelons que le lemme 2.2.1 borne, sous certaines circonstances, le rapport maximum entre les utilités de deux allocations. Malheureusement, ce rapport est aussi large que possible, puisqu'il atteint le quotient maximum entre les utilités de deux allocations.

Proposition 2.4.2

La perte maximum dans le pire cas pour le mécanisme de loterie est non bornée dans le cas général.

Dans le cas où $\varphi \neq 0$ et où le nombre de types k est constant, alors elle atteint la borne du lemme 2.2.1 : $\frac{1+\varphi}{\varphi} \cdot k$

Preuve. Pour prouver ce résultat, nous allons construire une famille d'instances $(I_k)_{k \in \mathbb{N}}$ telle que l'instance I_k ait k types et que le ratio entre les utilités de l'allocation optimale et de la pire allocation soit égal à la borne (et donc tende vers l'infini quand k tend vers l'infini). Considérons l'instance I_k pour $k \in \mathbb{N}$. Cette instance est constituée de k types de k agents chacun sauf pour le premier, et de k blocs de k objets (donc $\forall p \in [[2, k]]$, $|T_p| = k$ et $\forall q \in [[1, k]]$, $|B_q| = k$). Le premier type, en revanche, sera composé de $k^2 + k$ agents. Nous avons donc au total $|\mathcal{N}| = 2k^2$ agents et $|\mathcal{M}| = k^2$ objets. Nous définissons les quotas de la manière suivante :

- $\forall q \in [[1, k]]$, $\lambda_{1,q} = k$
- $\forall p \in [[2, k]]$, $q \in [[1, k]]$, $\lambda_{p,q} = 1$

Nous pouvons maintenant définir les utilités : les k^2 premiers agents de T_1 ont une utilité de 1 pour tous les objets, tandis que les k autres agents de T_1 ainsi que les agents des autres types sont neighbour-focused. Par soucis de clarté, dans la suite nous séparerons T_1 en deux sous-groupes, $T_{1,1}$ qui contient les k^2 agents de T_1 qui ont une utilité de 1 pour chaque objet, et $T_{1,2}$, qui contient les k agents neighbour-focused.

L'allocation A^* obtenue en faisant passer les agents de $T_{1,1}$ en premier dans le mécanisme de loterie (ce qui leur alloue tous les objets) est évidemment optimale pour le bien-être social utilitariste, puisque tous les agents qui ont reçu un objet auront une utilité de $1 + \varphi$. L'utilité globale vaut donc $U(A^*) = |\mathcal{M}|(1 + \varphi) = k^2(1 + \varphi)$.

Considérons maintenant l'allocation A_0 générée par le mécanisme de loterie appliqué à un ordre où les agents des groupes T_i , $i \neq 1$ sont placés en premier, puis les agents de $T_{1,2}$, puis les agents de $T_{1,1}$. Dans l'allocation résultante, tous les agents reçoivent un objet, sauf les agents de $T_{1,1}$ qui n'en reçoivent aucun. De plus, à cause des quotas, chaque bloc ne peut contenir qu'un seul agent de chaque type autre que T_1 , et comme les agents de T_1 choisissent en dernier, il ne leur restera qu'un seul objet libre dans chaque bloc. Par conséquent, chaque agent recevant un objet a une utilité de $\frac{\varphi}{k}$. Cette allocation a donc une utilité globale de $U(A_0) = \frac{|\mathcal{M}| \cdot \varphi}{k}$.

Le rapport entre les utilités des deux allocations vaut $\frac{U(A^*)}{U(A_0)} = \frac{1+\varphi}{\varphi} k$, ce qui est le ratio maximum possible.

De plus, si k n'est pas borné, alors ce rapport ne l'est pas non plus. □

Nous allons maintenant considérer comment le mécanisme de loterie interagit avec la notion de stabilité. De toute évidence, rien n'impose qu'une allocation produite par

ce mécanisme soit stable. Cependant, nous pouvons montrer quelque chose de plus précis : pour une instance donnée, il n'est pas garanti qu'il existe un ordre sur les agents tel qu'un mécanisme de loterie appliqué sur cet ordre renvoie une allocation stable.

Proposition 2.4.3

Il existe des instances pour lesquelles le mécanisme de loterie ne retourne jamais d'allocation stable, quel que soit l'ordre des agents.

Preuve. Pour prouver ce résultat, nous allons construire une telle instance.

Considérons une instance I avec deux types de deux agents chacun : $T_1 = \{a_1, a_2\}$ et $T_2 = \{a_3, a_4\}$, et deux blocs de deux objets chacun : $B_1 = \{h_1, h_2\}$ et $B_2 = \{h_3, h_4\}$. Nous posons les quotas suivant : $\lambda_{1,1} = \lambda_{2,2} = 1$, $\lambda_{2,1} = \lambda_{1,2} = 2$. En ce qui concerne les utilités, $\varphi = 1$, et les fonctions d'utilités de position sont telles que :

- $\forall a \in T_1$, $u_a^I(h_1) = 1$, and $u_a^I(h_2) = u_a^I(h_3) = u_a^I(h_4) = 0.75$
- $\forall a \in T_2$, $u_a^I(h_3) = 1$, and $u_a^I(h_1) = u_a^I(h_2) = u_a^I(h_4) = 0.75$

Lorsque l'on va appliquer le mécanisme de loterie sur cette instance, quelque soit le premier agent à choisir, il va choisir l'objet qui pour lui a une utilité de 1. Dans cette configuration, son utilité sera de $1 + 0.5$, et l'autre agent de son groupe devra choisir un bien de l'autre bloc, puisque que le premier agent remplit les quotas pour son type dans son bloc. Cependant, une fois les deux agents placés, le premier peut alors obtenir une utilité de $0.75 + 1$ en échangeant son bien pour être avec l'autre agent de son type, il cherchera donc à faire un échange. De même, les agents de l'autre type souhaiteront également se regrouper, l'un d'entre eux acceptera donc l'échange. L'allocation produite par le mécanisme ne sera donc jamais stable.

Pour mieux comprendre ce qui se passe, considérons un exemple : supposons que l'ordre soit (a_1, a_2, a_3, a_4) . L'agent a_1 va choisir en premier et prendre l'objet h_1 , pour une utilité de 1.5. L'agent a_2 est ensuite obligé de prendre un objet de B_2 (puisque a_1 remplit le quota $\lambda_{1,1}$). Celui qu'il choisit n'est pas important, supposons qu'il s'agisse de h_4 . C'est ensuite au tour de a_3 , qui va choisir h_3 , puis a_4 pour lequel il ne reste que h_2 . Les agents a_1 et a_3 ont alors des utilités de 1.5, et pourraient avoir des utilités de 1.75 s'ils échangeaient leurs objets, ce qui est permis par les quotas. L'allocation n'est donc pas stable. □

Nous pouvons également étudier les deux cas particuliers, celui où tous les agents sont neighbour-focused et celui où tous les agents sont item-focused.

Proposition 2.4.4

Le mécanisme de loterie ne retourne pas toujours une instance stable, même dans le cas où les agents sont neighbour-focused.

Il retourne toujours une instance stable dans le cas où les agents sont item-focused.

Proof. Commençons par le cas neighbour-focused. Dans ce cas, les objets auront tous la même utilité pour les premiers agents de chaque groupe, ils choisiront donc au hasard parmi eux. Considérons une instance I avec deux types de deux agents chacun : $T_1 = \{a_1, a_2\}$ et $T_2 = \{a_3, a_4\}$, et deux blocs de deux objets chacun : $B_1 = \{h_1, h_2\}$ et $B_2 = \{h_3, h_4\}$. Nous posons tous les quotas égaux à 2, ils ne restreindront donc pas les allocations. Si les deux premiers agents à choisir sont de type différent, puisque tous les objets leur sont indifférents, il est possible qu'ils choisissent tout deux des objets du même bloc, ce qui rend l'allocation résultante instable (puisque les agents d'un même type voudront se regrouper pour maximiser leur utilité).

Dans le cas item-focused, la preuve est immédiate, puisque les utilités des agents ne changent pas entre le moment où ils choisissent leur objet et la fin de l'exécution du mécanisme : dans une instance I quelconque, pour tout couple d'agents $(a_1, a_2) \in \mathcal{N}$, s'ils ont tous les deux reçu des objets et a_1 a choisi avant a_2 , alors a_1 préfère son objet à celui de a_2 . Par conséquent, il est impossible de trouver un échange valide améliorant : l'allocation est stable. \square

2.5 Procédure d'échanges bilatéraux

L'idée de départ dans la conception d'un nouveau mécanisme est de se concentrer sur la notion de stabilité. Même si la stabilité n'est pas nécessairement corrélée à l'optimalité, nous avons vu qu'il était possible de les concilier dans une certaine mesure. De plus, la stabilité a des avantages qui lui sont propres, par exemple, le fait que les agents ne puisse la modifier sans empirer leur situation peut garantir la longévité d'une allocation, elle apparaîtra également plus légitime.

Puisque la stabilité est une propriété qui se vérifie sur chaque couple d'agents, nous allons considérer une approche décentralisée, où chaque agents essaiera d'améliorer sa situation, en effectuant des échanges valides mutuellement améliorants avec d'autres agents jusqu'à ce que cela ne soit plus possible.

Nous définissons donc un nouveau mécanisme, appelé algorithme d'échanges bilatéraux, qui fonctionne en deux étapes : il va tout d'abord sélectionner une allocation valide quelconque, puis, tant que l'allocation n'est pas stable, il va appliquer un échange valide améliorant entre 2 agents. Ce mécanisme nécessite deux méthodes de sélection, une qui va sélectionner l'allocation valide initiale, et une qui va sélectionner quel échange appliquer dans le cas où plusieurs échanges valides augmentant sont possibles. Il est envisageable de définir différentes heuristiques pour ces méthodes de sélection. On peut par exemple effectuer en priorité les échanges qui augmentent

le plus le bien-être social utilitariste, ou l'utilité des agents impliqués, ou ceux qui augmentent l'utilité des agents les plus mal lotis. L'étude des heuristiques possibles n'entre pas dans le cadre de ces travaux, nous utiliserons une sélection aléatoire uniforme parmi les possibilités comme méthode par défaut.

Ce mécanisme est inspiré d'une catégorie spéciale de jeux de Schelling, les *swap schelling games*, qui établie une dynamique similaire entre ces agents (Chauhan *et al.* [2018]). On peut également le voir comme une version restreintes des cycles d'échanges utilisés par des travaux comme ceux de Beynier *et al.* [2019a] sur des problèmes d'allocations de ressources indivisibles classiques.

La première question soulevée par cet algorithme est de savoir s'il va forcément terminer. Heureusement, on peut prouver que c'est le cas.

Proposition 2.5.1: Terminaison

Dans le cas général, l'algorithme d'échanges bilatéraux finira forcément par atteindre un équilibre et terminer.

Preuve. Nous allons prouver que l'algorithme termine toujours en montrant qu'il s'agit d'un jeu de potentiel pour la fonction de potentiel suivante sur les allocations :

$$pot(A) = sw(A) + \sum_{i \in \mathcal{N}} u_i^I(A)$$

Pour montrer la terminaison, il suffit de montrer que si on peut passer d'une allocation A à une allocation A' par un échange valide améliorant, alors $pot(A) < pot(A')$.

D'après le lemme 2.3.1, nous savons que la variation d'utilité globale vaudra :

$$sw(A') - sw(A) = \Delta U = \Delta u^I + 2 \cdot \Delta u^N$$

Or, puisque l'échange est augmentant, les utilités individuelles des agents impliqués augmentent.

$$\text{Donc } \Delta u^I + \Delta u^N > 0$$

$$\text{D'où } 2 \cdot \Delta u^I + 2 \cdot \Delta u^N = \Delta u^I + \Delta U > 0$$

$$\text{Or, } pot(A') - pot(A) = sw(A') - sw(A) + \sum_{i \in \mathcal{N}} u_i^I(A') - \sum_{i \in \mathcal{N}} u_i^I(A)$$

Puisque les seules utilités de position qui changent lors de l'échange sont celles des agents impliqués, on peut simplifier l'équation : $\sum_{i \in \mathcal{N}} u_i^I(A') - \sum_{i \in \mathcal{N}} u_i^I(A) = \Delta u^I$

$$\text{Donc } pot(A') - pot(A) = sw(A') - sw(A) + \Delta u^I = \Delta u^I + 2 \cdot \Delta u^N + \Delta u^I > 0$$

Donc le potentiel $pot(A)$ augmente strictement lors d'un échange valide améliorant. Or, on sait que $pot(A) \leq |\mathcal{N}|2(1 + \varphi)$ (grâce aux bornes sur les utilités des agents). Par conséquent, l'algorithme d'échanges bilatéraux termine forcément et retourne une allocation stable. \square

Proposition 2.5.2

Le rapport entre les utilités de l'allocation optimale et d'une allocation retournée par l'algorithme d'échanges n'est pas borné.
Si le nombre de type d'agents k est fixé et $\varphi \neq 0$, alors il atteint la borne du lemme 2.2.1

Preuve. Nous considérons un paramètre $x \in \mathbb{N}$, et nous construisons une famille d'instances $(I_k)_{k \in \mathbb{N}}$ telles que l'instance I_k ait k types de $xk - 1$ agents chacun, et k blocs de $xk - 1$ objets chacun. Nous numérotions les objets et les agents par des triplets de $[[1, k]]^2 \times [[1, x]] \setminus \{(z, z, x) : \forall z \in [k]\}$ en respectant les contraintes suivantes :

- $\forall (p, q, y) \in [[1, k]]^2 \times [[1, x]] \setminus \{(z, z, x) : \forall z \in [k]\}, a_{(p,q,y)} \in T_p$
- $\forall (p, q, y) \in [[1, k]]^2 \times [[1, x]] \setminus \{(z, z, x) : \forall z \in [k]\}, h_{(p,q,y)} \in B_p$

Notons que $[[1, k]]^2 \times [[1, x]] \setminus \{(z, z, x) : \forall z \in [k]\} = xk^2 - k = k(xk - 1)$
Nous définissons les quotas de la manière suivante :

$$\forall (p, q) \in [[1, k]]^2, \lambda_{p,q} = \begin{cases} xk - 1 & \text{if } p = q \\ x & \text{otherwise} \end{cases}$$

Nous pouvons maintenant détailler les fonctions d'utilité des agents :

$$\forall (p, q, i) \in [[1, k]]^2 \times [[1, x]] \setminus \{(z, z, x) : \forall z \in [k]\}, u_{a_{p,q,i}}^I(A) = \begin{cases} 1 & \text{if } A(a_{p,q,i}) = h_{p,q,i} \\ 0 & \text{otherwise} \end{cases}$$

L'allocation optimale A^* est alors celle définie par $\forall (p, q, i) \in [[1, k]]^2 \times [[1, x]] \setminus \{(z, z, x)\}, A^*(a_{p,q,i}) = h_{p,q,i}$, c'est-à-dire l'allocation où tous les agents d'un même type sont dans le même bloc, et où tous les agents ont un objet d'utilité 1. Le bien-être social est alors de $sw(A^*) = |\mathcal{N}|(1 + \varphi) = k(xk - 1)(1 + \varphi)$.

Considérons d'un autre côté l'allocation A^\perp , que nous allons construire de manière à ce qu'elle soit stable et minimise l'utilité sur l'ensemble des allocations valides.

$$\forall p \in [[1, k]], y \in [[1, x - 1]], A^\perp(a_{(p,p,y)}) = \begin{cases} h_{(p,p,y-1)} & \text{si } y \in \{2, \dots, x - 1\}, \\ h_{(p,p,x-1)} & \text{si } y = 1 \end{cases}$$

Pour allouer les agents $a_{(p,q,y)}$ avec $p \neq q$, nous allons considérer k permutations $(\sigma_p)_{p \in [[1, k]]}$ vérifiant les propriétés suivantes :

- Pour tout $p \in [[1, k]]$, σ_p est une permutation sur $\{1, \dots, k\} \setminus \{p\}$
- $\forall p \in [[1, k]], q \in [[1, k]] \setminus \{p\}, \sigma_p(q) \neq q$
- $\forall p, p' \in [[1, k]]$ tels que $p \neq p', \forall q \in [[1, k]] \setminus \{p\}, \sigma_p(q) \neq \sigma_{p'}(q)$

Nous discuterons de comment construire une telle famille dans l'annexe A.
On utilise alors ces permutations pour compléter l'allocation en posant :

$$\forall p \in [[1, k]], q \in [[1, k]] \setminus \{p\}, y \in [[1, x]], A^\perp(a_{(p,q,y)}) = h_{(q, \sigma_p(q), y)}$$

Nous aurons alors, par construction, que chaque type d'agents T_p aura $x - 1$ agents dans le bloc B_p et x agents dans chaque autre bloc B_q . De plus, chaque agent sera alloué à un bien qui lui confèrera une utilité de zéro.

Cette allocation satisfait les quotas, elle est donc valide. Cependant, il nous reste à montrer qu'elle est stable.

Considérons un agent quelconque $a_{p,q,y}$ pour un certain (p, q, y) dans $[[1, k]]^2 \times [[1, x]] \setminus \{(z, z, x) : \forall z \in [k]\}$.

Il existe deux manières pour cette agents d'augmenter son utilité:

- Il peut augmenter son utilité de voisinage. Pour cela il doit se déplacer vers un bloc contenant plus d'agents que son bloc actuel. Il y a deux possibilités:
 - cas $p \neq q$: dans ce cas l'agent se trouve dans un bloc B_q avec x agents de type T_p . Le seul bloc pouvant accueillir plus que x agents de son type est le bloc B_p , mais ce dernier n'a à présent que $x - 1$ agents, par conséquent se déplacer dans ce bloc n'augmenterait pas l'utilité de voisinage.
 - cas $p = q$: dans ce cas l'agent se trouve dans le bloc B_p . Pour augmenter son utilité, il doit changer de bloc, mais comme les quotas pour son groupe sont remplis dans tous les autres blocs, il ne peut qu'échanger avec un autre agent du type T_p . Or, par construction, son objet a une utilité de zéro pour tout agent de T_p n'étant pas dans le bloc B_p , par conséquent aucun agent de son type présent dans un autre bloc ne souhaite échanger avec lui.

Aucun agent ne peut donc échanger son objet de sorte à augmenter son utilité de voisinage.

- il peut augmenter son utilité de position. Pour cela il doit parvenir à récupérer l'unique objet qui lui confère une utilité de 1. Il y a encore une fois deux possibilités:
 - cas $p = q$: dans ce cas l'objet lui conférant une utilité de 1 est possédé par l'agent $a_{p,p,y+1}$ (si $y < x - 1$) ou $a_{p,p,1}$ (sinon). Par construction, dès que $x > 3$, cet agent obtiendra une utilité de zéro de l'objet de l'agent $a_{p,p,y}$ et refusera donc d'échanger avec ce dernier.
 - cas $p \neq q$: dans ce cas l'objet $h_{p,q,y}$ lui conférant une utilité de 1 est possédé par l'agent $a_{p',p,y}$ avec p' tel que $\sigma_{p'}(p) = q$. Puisque nous savons qu'aucun agent ne peut augmenter son utilité de voisinage, le seul cas où l'agent $a_{p',p,y}$ accepterait d'échanger serait celui où l'objet $h_{q, \sigma_p(q), y}$ lui confère une utilité de 1, c'est-à-dire que $p' = q$ et $\sigma_p(q) = p$, ce qui est impossible (puisque la fonction σ_p ne peut retourner la valeur p). Donc cet agent refusera d'échanger.

Par conséquent, il n'y a pas d'échanges rationnels possibles: l'allocation est stable.

Nous calculons alors le bien-être social utilitariste dans l'allocation A^\perp . Chaque type T_p a $x-1$ agents dans le bloc B_p qui ont une utilité de $\frac{x-1}{xk-1}$ et x agents dans chacun des $k-1$ autres blocs qui ont une utilité de $\frac{x}{xk-1}$. Le total est donc:

$$sw(A^\perp) = |\mathcal{N}| \times 0 + \varphi k \left(\frac{(x-1)^2}{xk-1} + (k-1) \frac{x^2}{xk-1} \right)$$

$$sw(A^\perp) = \varphi \cdot k \frac{(x^2 - 2 \cdot x + 1) + x^2(k-1)}{xk-1}$$

Par conséquent, le rapport entre les valeurs minimum et maximum que l'utilité peut prendre est égale à:

$$\frac{sw(A^*)}{sw(A^\perp)} = \frac{(1+\varphi)(xk-1)k}{\frac{\varphi k(x^2k-2x+1)}{xk-1}} = \frac{(1+\varphi)(xk-1)^2}{\varphi(x^2k-2x+1)}$$

Quand x tends vers $+\infty$, ce rapport tends vers $\frac{(1+\varphi)(xk)^2}{\varphi x^2 k} = \frac{(1+\varphi)k}{\varphi}$, notre borne est donc atteinte à la limite. \square

Nous nous devons également de mentionner la notion de prix de l'anarchie, qui est une mesure courante de l'impact lié au fait de laisser les agents décider d'eux-mêmes plutôt que d'imposer une solution centralisée. Sa définition usuelle est le rapport entre les utilités de la solution optimale et du meilleur équilibre de Nash. Cependant, la notion d'équilibre de Nash est floue dans notre contexte, puisqu'un agent seul n'est pas autorisé à impacter l'allocation courante, il lui faut trouver un deuxième agent qui accepterait d'échanger son objet. Nous considérerons donc les allocations stables comme les équivalents des équilibres de Nash, et adaptons la définition en conséquence :

Définition 2.5.1: Prix de l'Anarchie

Le **Prix de l'Anarchie (PoA)** est défini comme le rapport maximum sur les instances I du problème entre l'utilité de l'allocation optimale A_I^* et l'utilité de la pire allocation stable A_I^\perp .

$$PoA = \sup_{\mathcal{I}} \frac{sw(A_{\mathcal{I}}^*)}{sw(A_{\mathcal{I}}^\perp)}$$

Nous pouvons directement déduire le prix de l'anarchie dans notre problème à partir du lemme 2.2.1 (qui nous dit qu'un tel ratio est non borné dans le cas général et majoré par $\frac{1+\varphi}{\varphi}k$ si k est fixé et $\varphi \neq 0$) et en utilisant la proposition 2.5.2 (qui nous dit que les retours possibles de l'algorithme d'échanges qui sont inclus dans l'ensemble des allocations stables atteignent cette borne). Par conséquent, nous avons le résultat suivant :

Corollaire 2.5.1

Le prix de l'anarchie du problème est non borné dans le cas général.

Si k est fixé et $\varphi \neq 0$, alors on a $PoA = \frac{1+\varphi}{\varphi}k$

2.6 Expérimentations

Nous allons maintenant parler des expériences effectuées dans l'objectif d'évaluer les performances pratiques des mécanismes que nous avons présentés. Nous considérerons 4 variantes de ces procédures :

- RSEQ : Les agents sont considérés dans un ordre aléatoire, et sélectionnent chacun à leur tour un objet aléatoirement (en respectant les contraintes de diversité). Cette approche sera utilisée comme un point de référence pour comparer les autres méthodes, et pour générer des allocations de manière complètement aléatoire.
- SEQ : Le mécanisme de loterie comme décrit dans la section 2.4. Les agents sont considérés dans un ordre aléatoire, et sélectionnent chacun leur tour le meilleur objet qui respecte les contraintes de diversité.
- RSEQ+SWAP : Une procédure en deux étapes consistant à générer une allocation avec RSEQ, puis à appliquer l'algorithme d'échanges bilatéraux à partir de cette allocation (tel que décrit dans la section 2.5).
- SEQ+SWAP : Une procédure en deux étapes consistant à générer une allocation avec SEQ, puis à appliquer l'algorithme d'échanges bilatéraux à partir de cette allocation (tel que décrit dans la section 2.5).

Nous allons mener trois séries d'expériences. La première concernera de petites instances (40 agents et 40 objets), sur lesquelles il est possible de calculer l'allocation optimale, afin de mesurer la distance à l'optimal des allocations retournées par les algorithmes. La deuxième série d'expérience considère les algorithmes appliqués au jeu de données de l'immobilier de Singapour, qui contient 1350 objets (les logements à répartir entre les demandeurs), pour comparer leurs performances. Enfin, lors de la troisième série, nous étudierons la complexité pratique de RSEQ+SWAP et SEQ+SWAP sur des instances de plus grandes tailles générées aléatoirement.

2.6.1 Prix de la stabilité empirique

Dans cette partie, nous allons considérer de petites instances pour lesquelles il est possible de calculer l'allocation optimale de manière exacte, afin de comparer cette dernière aux résultats que nous obtiendrons.

Il est possible de calculer l'allocation optimale grâce à un programme linéaire contenant un nombre de contraintes polynomial en la taille du problème. Ce programme linéaire est le suivant :

$$\max \sum_{i \in \mathcal{N}} \sum_{h \in \mathcal{M}} x_{ih} \times u_i^I(h) + \varphi \sum_{i \in \mathcal{N}} \sum_{j \in \mathcal{T}(i)} \sum_{q \in [l]} \frac{s_{ijq}}{|B_q|} \quad (2.1)$$

$$\text{s.t.} \quad \sum_{h \in \mathcal{M}} x_{ih} = 1 \quad \forall i \in \mathcal{N} \quad (2.2)$$

$$\sum_{i \in \mathcal{N}} x_{ih} = 1 \quad \forall h \in \mathcal{M} \quad (2.3)$$

$$\sum_{i \in T_p} \sum_{h \in B_q} x_{ih} \leq \lambda_{pq} \quad \forall p \in [k], q \in [l] \quad (2.4)$$

$$b_{iq} = \sum_{h \in B_q} x_{ih} \quad \forall i \in \mathcal{N}, q \in [l] \quad (2.5)$$

$$s_{ijq} \leq b_{iq} \quad \forall i, j \in \mathcal{N}, q \in [l] \quad (2.6)$$

$$s_{ijq} \leq b_{jq} \quad \forall i, j \in \mathcal{N}, q \in [l] \quad (2.7)$$

$$c_{ij1} \geq \sum_{h \in \mathcal{M}} (x_{jh} - x_{ih}) \times u_i^I(h) + \varphi \sum_{q \in [l]} \left(\frac{b_{jq} - s_{ijq}}{|B_q|} + \sum_{a \in \mathcal{T}(i) \setminus \{j\}} \frac{s_{jaq}}{|B_q|} - \sum_{a \in \mathcal{T}(i)} \frac{s_{iaq}}{|B_q|} \right) \quad \forall i, j \in \mathcal{N} \quad (2.8)$$

$$c_{ij2} \geq \sum_{h \in \mathcal{M}} (x_{ih} - x_{jh}) \times u_j^I(h) + \varphi \sum_{q \in [l]} \left(\frac{b_{iq} - s_{ijq}}{|B_q|} + \sum_{a \in \mathcal{T}(j) \setminus \{i\}} \frac{s_{iaq}}{|B_q|} - \sum_{a \in \mathcal{T}(j)} \frac{s_{jaq}}{|B_q|} \right) \quad \forall i, j \in \mathcal{N} \quad (2.9)$$

$$c_{ij3} \times \Lambda \geq \sum_{q \in [l]} \left(\lambda_{T(i)q} \times b_{jq} - \sum_{a \in T(i)} s_{jaq} \right) \quad \forall i, j \in \mathcal{N}, j \notin \mathcal{T}(i) \quad (2.10)$$

$$c_{ij4} \times \Lambda \geq \sum_{q \in [l]} \left(\lambda_{T(j)q} \times b_{iq} - \sum_{a \in T(j)} s_{iaq} \right) \quad \forall i, j \in \mathcal{N}, j \notin \mathcal{T}(i) \quad (2.11)$$

$$\sum_{r=1}^4 c_{ijr} \leq 3 \quad \forall i, j \in \mathcal{N} \quad (2.12)$$

$$x_{ih} \in \{0, 1\} \quad \forall i \in \mathcal{N}, h \in \mathcal{M} \quad (2.13)$$

$$b_{iq} \in \{0, 1\}, s_{ijq} \in \{0, 1\} \quad \forall i, j \in \mathcal{N}, q \in [l] \quad (2.14)$$

$$c_{ijr} \in \{0, 1\} \quad \forall i, j \in \mathcal{N}, r \in [4] \quad (2.15)$$

Dans ce programme linéaire :

- x_{ih} , $b_{i,q}$, s_{ijq} et c_{ijr} pour $i, j \in \mathcal{N}$, $q \in [I]$, et $r \in [4]$, sont les variables du programme.
- x_{ih} vaut 1 si et seulement si l'objet h est alloué à l'agent i , et 0 sinon
- $b_{i,q}$ indique si l'agent a_i est alloué à un objet du bloc B_q
- s_{ijq} indique si les agent a_i et a_j sont tous les deux alloués à des objets du bloc B_q .
- c_{ij1} , c_{ij2} , c_{ij3} , c_{ij4} représente 4 conditions pour qu'il existe un échange valide améliorant entre les agents i et j . Il existe un tel échange si et seulement si les 4 conditions sont réunies.
- Λ est une constante définie par : $\Lambda = \max_{p \in [k], q \in [I]} \lambda_{pq}$
- la ligne 2.2 impose que chaque agent soit alloué à exactement 1 objet.
- la ligne 2.3 impose que chaque objet soit alloué à exactement 1 agent.
- la ligne 2.4 impose le respect des quotas de diversité.
- la ligne 2.5 impose que $b_{i,q}$ soit égale à 1 si et seulement si l'agent a_i est alloué à un objet du bloc B_q , et 0 sinon.
- les lignes 2.6 et 2.7 imposent que s_{ijq} soit égale à 1 si et seulement si les agent a_i et a_j sont tous les deux alloués à des objets du bloc B_q , et 0 sinon.
- la ligne 2.8 vérifie si l'agent a_i augmenterait son utilité en échangeant avec l'agent j .
- la ligne 2.9 vérifie si l'agent a_j augmenterait son utilité en échangeant avec l'agent i .
- la ligne 2.10 vérifie si les quotas autorisent l'agent a_i à prendre l'objet de l'agent a_j
- la ligne 2.11 vérifie si les quotas autorisent l'agent a_j à prendre l'objet de l'agent a_i
- la ligne 2.12 vérifie que les conditions pour qu'il existe un échange augmentant valide entre les agents a_i et a_j ne soient pas toutes vérifiées en même temps (chaque condition remplie vaut 1, cette ligne fait la somme et la contraint à être inférieure ou égale à 4).

Nous considérerons 20 instances générées aléatoirement contenant chacun 40 objets répartis en 5 blocs de taille égale, et 40 agents répartis en 5 types de tailles égale. Les quotas sont tels que : $\forall p, q \in [[1, 5]]^2$, $\lambda_{p,q} = 2$. Les utilités de position sont tirées au

hasard sur l'intervalle $[0, 1]$ suivant une probabilité uniforme. Nous considérons deux valeurs de compromis d'utilité φ : 0.5 et 1.

Sur chaque instance, nous avons calculé le prix de la stabilité, qui s'est avéré être toujours égal à 1, ce qui veut dire que sur toutes les instances, l'allocation d'utilité maximale est stable. La table 2.1 rapporte les pertes d'utilité relatives de chaque algorithme (définies comme le rapport entre l'utilité de l'allocation optimal et de l'utilité moyenne des allocations retournées par un algorithme).

Notons que les algorithmes RSEQ et SEQ ne retournent pas toujours des allocations valides et doivent être exécutés en moyenne 1.87 fois (± 1.74) pour RSEQ et 1.83 fois (± 1.29) pour SEQ avant que l'on atteigne une allocation stable.

	RSEQ	SEQ	RSEQ+SWAP	SEQ+SWAP
$\varphi = 0.5$	1.757 ± 0.144	1.073 ± 0.028	1.101 ± 0.023	1.069 ± 0.029
$\varphi = 1$	1.695 ± 0.118	1.084 ± 0.027	1.102 ± 0.026	1.074 ± 0.021

Table 2.1: Perte d'utilité moyenne sur de petites instances

Dans ce tableau, nous observons que RSEQ produit des résultats significativement moins efficaces que les trois autres algorithmes, qui présentent des performances similaires. En particulier, les pertes d'utilité sont très proches de 1, qui est le prix de la stabilité empirique, et très loin de la borne théorique sur les pertes énoncée dans le lemme 2.2.1, qui vaut $\frac{1+\varphi}{\varphi} \cdot k$ (donc 10 pour $\varphi = 1$ et 15 pour $\varphi = 0.5$).

On peut cependant remarquer que les pertes d'utilité de SEQ et de SEQ+SWAP augmentent avec φ , alors que celles de RSEQ+SWAP restent constantes. Cela pourrait être dû au fait que les premiers agents à choisir un objet le font sans connaissance de leurs futurs voisins, et que s'ils choisissent un objet de grande valeur, ils auront plus de réticence à se rapprocher des autres agents de leur groupe (augmentant ainsi leur utilité au détriment de celle de leurs voisins).

2.6.2 Comparaison des résultats sur les données réelles de l'immobilier de Singapour

Dans cette partie, nous utilisons un jeu de données réelles de l'immobilier de Singapour (décrit par Benabbou *et al.* [2018]). Ce jeu de données est composé de 1350 objets (des logements) séparés en blocs de taille 128, 162, 156, 249, 108, 94, 104, 190, et 159, et d'un ensemble d'agents \mathcal{N} (des demandeurs de logements) de taille $|\mathcal{N}| = 2700$, répartis en trois types T_1 , T_2 et T_3 tels que $|T_1| = 0.74|\mathcal{N}|$, $|T_2| = 0.13|\mathcal{N}|$ et $|T_3| = 0.13|\mathcal{N}|$. Les quotas de diversité sont définis par : $\forall q \in [1, l]$, $\lambda_{1,q} = 0.87 \times |B_q|$, $\lambda_{2,q} = 0.25 \times |B_q|$, $\lambda_{3,q} = 0.15 \times |B_q|$.

En ce qui concerne les utilités, nous posons $\varphi = 1$, et nous considérons deux modèles d'utilité de position :

- Un modèle basé sur la distance, que nous nommerons $\text{Dist}(\sigma^2)$, dans lequel chaque

agent a une position préférée dans $[0, 1]^2$, et calcule l'utilité qu'un logement lui procure à partir de la distance de ce logement à sa position préférée (les positions des logements sont celles des blocs qui les contiennent, qui sont décrites dans le jeu de données et ramenée sur $[0, 1]^2$). Plus précisément, pour un agent i et un objet h du bloc B_q , l'utilité de position $u_i^I(h)$ est tirée aléatoirement en suivant une distribution de probabilité normale $\mathcal{N}(1/d(p_i, L(B_q)), \sigma^2)$, où $L(B_q)$ est la position de B_q et $d(\cdot)$ est la distance Euclidienne.

- Un modèle basé sur l'ethnie (représentée par le type dans le jeu de données), que nous nommerons (Ethn(σ^2)), similaire au précédent à l'exception que tous les agents d'un même type auront la même position préférée.

En ce qui concerne la variance σ utilisée dans les lois normales ci-dessus, nous considérerons deux valeurs : 0.01 and 1.

	Dist(σ^2)		Ethn(σ^2)	
	$\sigma^2 = 0.01$	$\sigma^2 = 1$	$\sigma^2 = 0.01$	$\sigma^2 = 1$
RSEQ	903 ± 14	1465 ± 43	1133 ± 84	1466 ± 41
SEQ	989 ± 23	1678 ± 60	1264 ± 99	1706 ± 60
RSEQ+SWAP	1057 ± 19	1659 ± 58	1242 ± 93	1684 ± 55
SEQ+SWAP	1072 ± 23	1690 ± 61	1268 ± 101	1716 ± 61

Table 2.2: Valeurs moyennes du bien-être social pour le jeu de données de l'immobilier de Singapour

La table 2.2 rapporte les valeurs de bien-être social utilitariste des allocations retournées par les algorithmes appliqués sur les données de Singapour avec les utilités décrites. Chaque combinaison d'algorithme, de modèle d'utilité et de valeur de σ , a été appliquée sur 30 instances différentes. Notons qu'il ne nous est pas possible ici de calculer les pertes d'utilité, puisque le jeu de données est trop gros pour utiliser le programme linéaire.

Nous remarquons sans surprise que RSEQ reste moins performant que SEQ, RSEQ+SWAP, et SEQ+SWAP en termes d'utilité globale finale. Les résultats de ces trois derniers algorithmes restent relativement similaires, sauf dans le cas du modèle d'utilité basé sur la distance pour $\sigma^2 = 0.01$ (Dist(0.01)), où SEQ est beaucoup moins efficace que RSEQ+SWAP et SEQ+SWAP. Cela peut s'expliquer par le fait que les agents d'un même groupe auront des préférences (fonctions d'utilités) très différentes, et vont donc se disperser lors des premières affectations. SEQ+SWAP, qui donne aux agents la possibilité de se rapprocher une fois l'allocation initiale complétée, corrige cette tendance et présente donc une bien meilleur utilité globale.

2.6.3 Nombre d'échanges avant terminaison

Dans cette partie, nous analysons la complexité pratique de l'algorithme d'échange, c'est-à-dire le nombre d'échanges bilatéraux qui devront être effectués en moyenne

pour atteindre une allocation stable (et donc l'arrêt de l'algorithme).

Pour cela, nous allons considérer des instances générées aléatoirement pour lesquelles les tailles des ensembles d'agents et d'objets iront de 100 à 1400 par pas de 100. L'ensemble des objets sera divisé en $l = 10$ blocs de taille égale, et l'ensemble des agents en $k = 3$ types de tailles respectives $|T_1| = 0.74 \times |\mathcal{N}|$, $|T_2| = 0.13 \times |\mathcal{N}|$, et $|T_3| = 0.13 \times |\mathcal{N}|$. Les quotas de diversité seront tels que $\forall q \in \llbracket 1, 10 \rrbracket$, $\lambda_{1,q} = 0.87 \times |B_q|$, $\lambda_{2,q} = 0.25 \times |B_q|$, $\lambda_{3,q} = 0.15 \times |B_q|$. (Ces paramètres sont semblables à celui du jeu de données de Singapour.)

Les utilités de position seront tirées aléatoirement de manière uniforme sur l'intervalle $[0, 1]$, et nous posons $\varphi = 1$.

Pour chaque jeu de paramètres, nous effectuons 20 expériences.

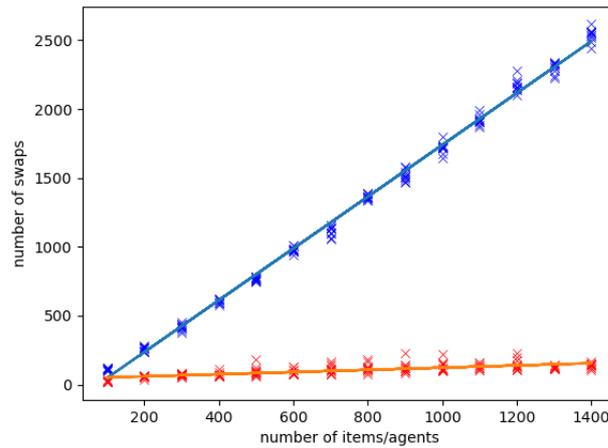


Figure 2.2: Régression linéaire sur le nombre d'échanges d'une exécution

Dans la figure 2.2, nous présentons une régressions linéaire sur les nombres d'échanges valides augmentant requis par l'algorithme pour atteindre un équilibre. Les résultats présentés sont :

- **En bleu : RSEQ+SWAP** nous obtenons une pente de 1,88 et une ordonnée à l'origine de -141,20, pour un facteur de corrélation de 0.988 (très forte corrélation).
- **En rouge : SEQ+SWAP** nous obtenons une pente de 0,08 et une ordonnée à l'origine de 43,53, pour un facteur de corrélation de 0.74 (forte corrélation).

Nous remarquons que le nombre d'échanges avant stabilité croît de manière linéaire sur les exemples pratiques. Des deux algorithmes, *SEQ + SWAP* termine beaucoup plus rapidement, ce qui s'explique facilement par la plus grande utilité de départ des agents, qui limite les gains possibles lors d'un échange. On peut de plus remarquer que le nombre d'échanges avant terminaison croît très lentement avec la taille de l'instance, ce qui laisse penser que ce mécanisme n'est pas très difficile à mettre en place même à grande échelle. Il nous faut cependant nous rappeler que l'application de ce mécanisme nécessite un nombre quadratique d'opérations entre chaque échange, puisqu'il nous

faut tester chaque paire d'agents. L'intérêt de cette méthode réside donc plutôt dans le fait de permettre aux agents d'améliorer leur situation en cherchant des échanges, puisqu'un agent peut vérifier avec qui il souhaite échanger en temps linéaire.

2.7 Conclusion du chapitre

Nous avons considéré dans ce chapitre un modèle qui oppose deux tendances opérant à deux échelles différentes. La première, issue du modèle de base, est une volonté de la société de garantir une certaine diversité, qui se traduit par des contraintes sous la forme de quotas. La deuxième, inspirée des jeux de Schelling, est une propension de la part des agents qui les pousse à chercher la proximité de leur semblables, représenté par une composante d'homophilie dans leurs fonctions d'utilité. Puisque nous introduisons des dynamiques propres aux jeux de Schelling, certaines problématiques de ce modèle deviennent pertinentes pour le nôtre, notamment les questions de stabilité. Nous avons donc étudié l'impact sur le bien-être social utilitariste de la recherche de stabilité et établi une borne sur la perte engendrée. Nous nous sommes ensuite intéressés aux mécanismes de génération d'allocation, le mécanisme de loterie d'abord, simple mais dont les résultats présentent peu de propriétés intéressantes. Puis le mécanisme d'échanges, plus long mais apportant des garanties de stabilité. Nous avons cependant vu qu'aucun de ces mécanismes ne parvient à s'éloigner de la borne sur la perte maximum. Enfin, nous avons effectué des expérimentations pour étudier les différents mécanismes, sur des instances générées aléatoirement et sur des données réelles. Il en est ressorti que le mécanisme d'échange peut être appliqué sur une allocation instable générée par une autre méthode sans crainte de perte d'efficacité, et qu'il est rapidement capable d'atteindre une allocation stable.

Les possibilités d'amélioration que pourraient apporter de prochains travaux concernent en premier lieu le raffinement du résultat de la proposition 2.3.2 : nous avons conjecturé que le prix de la stabilité est proche de 2, mais il nous faut encore trouver une preuve, qui consisterait en une famille d'instances pour laquelle le ratio entre l'utilité de l'allocation optimale et celle de la meilleure allocation stable tend vers 2. Une extension possible du modèle permettant plus de flexibilité dans sa représentation serait de considérer le cas de compromis d'utilité φ différents. Cela permettrait aux agents d'avoir différentes opinions sur l'importance relative de la composition du voisinage par rapport à la qualité de leurs allocations. Cela demanderait cependant de nouvelles preuves pour le prix de la stabilité et pour la terminaison de l'algorithme d'échanges bilatéraux (des expériences suggèrent que ce dernier se termine tout de même dans le cas de φ différents). Une troisième piste serait la question de la sélection de l'échange à effectuer quand plusieurs options sont possibles durant une itération de l'algorithme des échanges bilatéraux. Quelques expériences ont été effectuées pour tester une heuristique maximisant le gain d'utilité, les résultats suggéraient peu de différence en termes d'allocation finale, et les gains en nombre de swaps sont contrebalancés par les temps de calculs plus long pour chaque itération. Cependant, des heuristiques plus rapides à calculer peuvent être considérées. De plus, dans le cadre

d'une sélection aléatoire, les seuls résultats de complexité sont expérimentaux, nous n'avons pas de borne théorique sur le nombre d'itérations, il peut donc être pertinent d'étudier le comportement de cette heuristique dans le pire cas. Une quatrième piste serait de n'autoriser des échanges qu'entre des blocs géographiquement proches. Dans ce modèle, les positions des différents blocs induiraient une structure de graphe, où les sommets seraient les blocs et les arêtes relieraient deux sommets si les blocs correspondant sont suffisamment proches pour un échange. À noter qu'il existe déjà de nombreuses études sur des modèles où les agents appartiennent à un graphe et ne peuvent envier que leur voisins, tant dans le cadre du partage de gâteau (Abebe *et al.* [2017], Bei *et al.* [2017]) que dans le cadre de l'allocation de ressources indivisibles (Bredereck *et al.* [2022], Beynier *et al.* [2019b]). Cependant, nous nous distinguerions de ces travaux par une différence majeure: les relations de voisinage ne seraient pas statique mais dépendante de l'objet possédé. Pour le cas de l'immobilier de Singapour, des expériences seraient facile à mettre en oeuvre puisque le jeu de données comprend les positions des différents blocs. Pour le cas général, la question se pose de l'impact de la topographie du graphe des blocs sur la dynamique des échanges améliorants.

Enfin, une dernière piste serait de considérer des échanges impliquant plus que deux agents. Il s'agit là d'une extension naturelle des échanges bilatéraux, et il nous faut noter que Damamme *et al.* [2015] considèrent ces derniers comme un cas particulier d'un cycle d'échange à k agents, pour $k = 2$. L'intérêt de considérer une dynamique de k -cycle plutôt que d'échange bilatéraux est qu'il s'agit d'une dynamique très proche et donc partageant un certain nombre de propriété, cependant, en fonction de k , elle permet des échanges de plus ou moins large envergure ce qui impacte la notion de stabilité, on peut donc se demander si le prix de la stabilité va dépendre de k .

Les travaux présentés dans cette partie ont donné lieu à deux publications : un abstract étendu à la conférence AAMAS (Gross-Humbert *et al.* [2021]) et un article dans le journal TEAC (Transactions on Economics and Computation) (Gross-Humbert *et al.* [2022]).

Chapitre 3

Envie entre types d'agents

3.1 Introduction

L'objectif de ce chapitre est de trouver une manière de mesurer l'équité des allocations dans le cadre du problème de *house allocation* et dans lequel l'ensemble des agents est partitionné en types. Ce chapitre fait suite au précédent et considère un modèle obtenu en simplifiant celui de l'immobilier de Singapour. Nous souhaitons nous assurer que les allocations soient équitables non seulement à l'échelle des agents individuels mais aussi à l'échelle des types, et ce utilisant le concept de l'envie.

L'utilisation de la notion d'envie pour comparer des groupes d'agents est une question vaste sur laquelle de nombreuses approches ont été envisagées. C'est une question sur laquelle se penchent naturellement les travaux qui utilisent des types d'agents, comme ceux de Benabbou *et al.* [2018] ou ceux de Scarlett *et al.* [2023], qui essayent chacun d'y répondre à leur manière. Mais même des travaux dont les modèles n'ont pas de partition a priori des agents en groupes cherchent parfois à comparer des groupes quelconques d'agents entre eux. Par exemple, Aleksandrov et Walsh [2018a] cherchent à s'assurer que toute paire de deux groupes d'agents quelconques soit sans-envie. Cependant, chacun de ces travaux propose une approche différente pour adapter la notion d'envie à des groupes de tailles potentiellement très différentes. De manière générale, depuis l'introduction de la notion d'envie dans la littérature (par Foley [1967]), peu de travaux se penchent sur le comportement que l'on attendrait d'une mesure d'envie entre groupes d'agents.

L'un des articles cherchant à axiomatiser des propriétés que l'on attendrait d'une notion d'envie est celui de Bosmans et Öztürk [2018], qui pose un certain nombre de propriétés qu'une mesure numérique d'envie devrait suivre, telles que la normalisation, l'anonymat ou la décomposabilité. Cependant la pertinence de ces travaux reste limitée pour notre domaine. En effet, on considérera rarement l'envie sous la forme d'une valeur numérique à minimiser, on préférera chercher des seuils à ne pas dépasser. De plus, l'article suppose des préférences strictement monotones sur les lots d'objets, ce qui ne sera généralement pas le cas dans le cadre des problèmes de *house allocation*, et certains de ses axiomes ne s'appliquent plus sans cette hypothèse (la proportionnalité

notamment).

Un autre article discutant d'axiomatique dans le cadre des problèmes d'allocation est celui de Thomson [2019]. Cet article ne présente aucun axiome à proprement parler, mais propose une série de catégories et de qualificatifs se référant aux axiomes existants, afin de proprement classer ces derniers.

Notre approche va consister dans un premier temps à mettre en place un certain nombre d'axiomes simples adaptés à notre modèle, puis à considérer les définitions d'envie entre groupes utilisées dans la littérature au travers de ces axiomes, pour enfin construire une notion d'envie qui satisfait nos exigences. Nous terminerons par une étude plus en détails de la notion de monotonie.

3.2 Allocations de ressources entre groupes d'agents

Notre modèle sera très semblable à celui vu en dans la section 2.2. Toutefois, pour nous concentrer sur la question de l'équité entre groupes d'agents, une partie du modèle sera ramenée à un problème d'allocation de ressources classique : nous conservons les types d'agents, mais nous n'aurons plus de blocs d'objets, et par conséquent ni quotas ni utilité de voisinage. En revanche, comme les groupes d'agents (que nous appellerons types dans la suite) deviendront l'élément central du modèle, nous adapterons les notions d'utilité et d'allocation pour leur conférer une plus grande importance.

Définition 3.2.1: House allocation avec types d'agents

Une instance I du modèle contient les éléments suivant :

- un ensemble \mathcal{N} d'agents $\mathcal{N} = \{1, 2, 3 \dots n\}$ partitionnés en k types $\mathcal{T} = \{T_1, T_2, \dots T_k\}$,
- un ensemble \mathcal{M} d'objets $\mathcal{M} = \{1, 2, 3 \dots m\}$,
- pour chaque agent $i \in \mathcal{N}$, une fonction d'utilité $u_i : \mathcal{M} \rightarrow \mathbb{R}$ représentant les préférences de l'agent i sur les objets.

Nous allons aborder les allocations d'une manière différente du chapitre précédent. Nous allons maintenant chercher à modéliser les situations dans lesquelles l'autorité centrale ne peut allouer les ressources qu'aux types, tandis que les types disposent de mécanismes internes pour répartir leurs objets entre leurs agents.

Imaginons par exemple qu'un organisme de recherche, constitué de plusieurs équipes de recherche elles-même constituées de chercheurs, ait un certains nombre de bureaux à répartir. Dans ce cas, l'organisme va seulement attribuer les bureaux à ses équipes, et les équipes vont ensuite les attribuer à des chercheurs précis, selon une logique de répartition interne à l'équipe.

Un autre exemple est le problème d'allocation de ressources médicales : un gouvernement veut répartir du matériel médical entre ses hôpitaux, et chaque hôpital va

ensuite répartir le matériel qu'il a reçu entre ses différents services.

Nous allons donc considérer des allocations à deux niveaux : au niveau du type, nous allouons des lots d'objets à chaque type, puis au niveau individuel, chaque type répartira son lot d'objets entre ses agents. Ainsi, même si une allocation va appairer les agents et les objets deux à deux, nous considérerons que nous ne pourrions choisir directement seulement la répartition des objets entre les types. Chaque type disposant de son propre mécanisme d'allocation interne afin d'allouer un objet à un agent précis.

Nous commençons ainsi par définir les allocations sur les types.

Définition 3.2.2: Allocations aux types

Nous définissons une allocation A comme une partition de \mathcal{M} en k lots d'objets $A = A_1 \cup A_2 \cup \dots \cup A_k$ tels que $\forall p \in \llbracket 1, k \rrbracket$, A_p soit le lot d'objets alloué au type T_p . Nous imposons $\forall p \in \llbracket 1, k \rrbracket$, $|A_p| \leq |T_p|$

Nous allons maintenant nous intéresser à la répartition des objets entre les agents même d'un type. Nous n'avons pas pour l'instant défini la manière dont les types vont allouer leurs objets à leurs agents. Il est ici possible de choisir n'importe quelle méthode d'allocation, et même une méthode différente pour chaque type. Nous supposons dans ce qui suit que les types utilisent une méthode d'allocation cherchant à maximiser le bien-être utilitariste des agents au sein du type.

Définition 3.2.3: Allocations individuelles

Nous étendons toute allocation $A = (A_p)_{p \in \llbracket 1, k \rrbracket}$ en une fonction $\mathcal{N} \rightarrow \mathcal{M} \cup \{\emptyset\}$ définie telle que A maximise la valeur $\sum_{i \in \mathcal{N}} u_i(A(i))$ sous contrainte $\forall p \in \llbracket 1, k \rrbracket$, $\bigcup_{i \in T_p} A(i) = A_p$

Nous aurons dans la suite besoin que les types aient une manière de comparer les lots d'objets entre eux, nous allons donc définir une notion d'utilité sur les types qui représentera la qualité d'un lot par une valeur. Puisque tout lot d'objets qui sera alloué à un type sera réparti de manière à maximiser le bien-être social utilitariste des membres de ce type, il est raisonnable de considérer qu'un type mesurera la qualité d'un lot d'objets par le bien-être social utilitariste qu'il peut lui procurer. Notre fonction d'utilité va donc mesurer le bien-être que le type obtiendrait d'un lot si il allouait les objets qu'il contient à ses agents de manière à maximiser le bien-être utilitariste (le mécanisme d'allocation interne choisi pour notre cadre).

Formellement :

Définition 3.2.4: Utilité de type

Soit une instance I contenant les types d'agent T_1, \dots, T_k .

L'**utilité de type** $V_p(L)$ d'un type T_p (pour $p \in \llbracket 1, k \rrbracket$) pour un lot d'objets L est définie par :

$$V_p(L) = \begin{cases} \max_{X \in \mathcal{X}_L} \sum_{i \in T_p} \sum_{j \in L} u_i(j)x_{ij}, & \text{si } L \neq \emptyset; \\ 0, & \text{sinon.} \end{cases}$$

Où:

- $u_i(j) \in \mathbb{R}_+$ est l'utilité de l'agent $i \in T_p$ pour l'objet $j \in L$
- \mathcal{X}_L est l'ensemble des matrices binaires $X = (x_{ij})_{i \in T_p, j \in L}$ qui représentent les allocations des objets de L aux agents de T_p , de la manière suivante:

$$x_{ij} = \begin{cases} 1 & \text{si } j \in L \text{ est alloué à } i \in T_p; \\ 0 & \text{sinon} \end{cases}$$

Notons que cette fonction d'utilité est sous-modulaire¹, mais n'est pas additive. Notons également qu'elle peut être calculée en temps polynomial puisqu'elle peut être réduite à un problème de couplage de poids maximum classique.

Nous souhaitons donc créer des allocations à l'échelle des types qui soient équitables, ce qui requiert tout d'abord de définir la notion d'équité. Une notion très présente dans la littérature pour définir l'équité est la notion d'envie, vers laquelle nous allons nous orienter. La pertinence de cette notion dans notre contexte vient du fait qu'elle soit calculée individuellement par les agents, sans consulter d'autorité centrale, elle se place donc dans la continuité du chapitre précédent. Elle présente également d'autres avantages: un agent peut déterminer si il est envieux en utilisant uniquement ses propres connaissances, elle ne requiert donc pas de communication. Elle représente de plus une propension naturelle des agents à se comparer entre eux.

L'usage de la notion d'envie, dont il existe plusieurs versions alternatives, consiste à établir des conditions dans lesquelles un agent (ou dans notre cas, un type d'agent) ne serait pas "envieux" d'un autre. On considère alors qu'une allocation est équitable lorsqu'elle ne contient pas d'envie. Comme il existe des cas où cela ne sera pas possible, on se tournera parfois vers des relaxations de l'envie.

Depuis son introduction par Foley [1967], la notion d'envie a été étudiée par de très nombreux travaux, comprenant par exemple Bouveret et Lemaître [2015], Benabbou *et al.* [2020], Budish [2011], Caragiannis *et al.* [2019], ou encore Plaut et Roughgarden

¹Une fonction d'utilité u est sous-modulaire si pour tout ensemble d'objets E_1 et E_2 , elle vérifie $u(E_1) + u(E_2) \geq u(E_1 \cup E_2) + u(E_1 \cap E_2)$

[2018] . Cependant, il existe peu de travaux se plaçant dans le cadre de types d'agents, et la plupart des travaux considérant des groupes d'agents autorisent les agents à former des groupes arbitraires (comme Aleksandrov et Walsh [2018a] et Conitzer *et al.* [2019]).

Par conséquent, notre objectif ici est de trouver une notion d'envie appropriée à notre cas, c'est-à-dire à un modèle où l'ensemble des agents est partitionné en un ensemble de type et où chaque agent ne peut utiliser qu'un seul objet. Nous allons commencer par établir un certain nombre d'axiomes qui devront être satisfait par cette notion.

3.3 Axiomatisation des notions d'envie

3.3.1 Définitions des axiomes

En premier lieu, avant de chercher une notion d'envie que l'on considère appropriée, il nous faut d'abord définir ce que l'on attend d'une telle notion. Pour cela, nous allons définir un ensemble d'axiomes, puis nous chercherons à définir une notion d'envie qui vérifie tous ces axiomes. Chacun de ces axiomes va considérer un cas spécifique (généralement assez extrême) dans lequel ce qu'on attend de l'envie est évident, par conséquent le non-respect de ces axiomes est signe d'un comportement incohérent.

Le premier axiome que nous allons considérer va nous servir à exprimer explicitement que la notion d'envie que nous souhaitons construire doit être une extension de la notion d'envie classique aux cas des groupes d'agents de tailles différentes. Cela signifie que dans le cas particulier où deux types sont de même taille, notre notion d'envie doit être équivalente à l'envie classique. Nous posons donc :

Axiome 3.3.1: Extension de la notion d'envie classique

Soit deux types T_p et T_q dans l'allocation A .
Si $|T_p| = |T_q|$, alors T_p envie T_q si et seulement si $V_p(A_p) < V_p(A_q)$

Les deux axiomes suivants que nous allons considérer transcrivent l'idée que l'opinion du type ne doit pas contredire une opinion unanime parmi ses agents.

Le premier axiome considère le cas extrême où chaque agent d'un type envie individuellement chaque agent d'un autre type. Intuitivement, si tous les agents s'accordent sur un manque d'équité omniprésent, le type composé de ces agents ne doit pas considérer l'allocation comme équitable. Par conséquent, on attendra que ce type soit en-vieux dans ce cas. On peut penser cette propriété comme un "pire cas" où n'importe quelles comparaison individuelle conduit à de l'envie.

Axiome 3.3.2: Envie par consensus

Soit deux types T_p et T_q dans l'allocation A .
 Si $\forall i \in T_p, j \in T_q$ on a $u_i(A(i)) < u_i(A(j))$, alors T_p envie T_q .

Un autre axiome que nous souhaitons considérer suit l'idée qu'un type ne devrait en envier un autre que si il est possible d'améliorer son allocation en utilisant les biens de l'autre type. Concrètement, nous considérerons le cas où un type obtiendrait tous les objets de l'autre type sans perdre un seul des siens. En effet, il s'agit du meilleur cas possible puisque l'utilité est monotone. Si dans ce cas, l'utilité du type recevant les objets est identique à celle qu'il obtenait précédemment, alors il ne doit pas être envieux.

Axiome 3.3.3: Non-envie par indifférence

Soit deux types T_p et T_q dans l'allocation A .
 Si $V_p(A_p \cup A_q) = V_p(A_p)$, alors T_p ne doit pas envier T_q .

Notons que puisque notre fonction d'utilité est le maximum du bien-être social utilitariste, le fait que l'utilité de l'union soit égale à celle du lot d'origine implique qu'aucun agent de T_p ne parvient à augmenter son utilité quand on lui donne accès au lot de T_q . Par conséquent, cet axiome est équivalent à dire que si aucun agent de T_p n'envie individuellement aucun agent de T_q , alors T_p ne doit pas envier T_q .

Proposition 3.3.1: Définition alternative de l'axiome 3.3.3

Dans notre cadre, une notion d'envie vérifie l'axiome 3.3.3 si et seulement si elle vérifie aussi la propriété suivante :
 Soit deux types T_p et T_q dans l'allocation A .
 Si $\forall i \in T_p, j \in T_q$ on a $u_i(A(i)) \geq u_i(A(j))$, alors T_p ne doit pas envier T_q .

Preuve. Pour l'implication : si il existe un agent $i \in T_p$ et un bien $h \in A_q$ tel que $u_i(h) > u_i(A(i))$, alors $U_{T_p}(A_p \cup \{h\}) \geq U_{T_p}(A_p) + u_i(h) - u_i(A(i))$ (si on rajoute h au lot de T_p , l'utilité dans la nouvelle allocation d'utilité maximum est au moins aussi grande que celle où le type alloue h à i). Or, $U_{T_p}(A_p) + u_i(h) - u_i(A(i)) > U_{T_p}(A_p)$, et comme l'utilité de type est monotone et $A_p \cup \{h\} \subseteq A_p \cup A_q$, on a $U_{T_p}(A_p \cup \{h\}) \leq U_{T_p}(A_p \cup A_q)$. Donc si il existe un agent $i \in T_p$ et un bien $h \in A_q$ tel que $u_i(h) > u_i(A(i))$, alors $U_{T_p}(A_p \cup A_q) > U_{T_p}(A_p)$, d'où par contraposée si $U_{T_p}(A_p \cup A_q) = U_{T_p}(A_p)$ alors $\forall i \in T_p, j \in T_q$ on a $u_i(A(i)) \geq u_i(A(j))$.

Pour la réciproque : on a immédiatement que si $\forall i \in T_p, j \in T_q$ on a $u_i(A(i)) \geq u_i(A(j))$, alors l'allocation maximisant le bien-être social utilitariste pour le lot $A_p \cup A_q$ n'a pas besoin d'utiliser de biens de A_q , par conséquent c'est la même que pour le lot A_p , et a donc la même utilité. Donc $U_{T_p}(A_p \cup A_q) = U_{T_p}(A_p)$. □

Les deux précédents axiomes concernent des cas extrêmes où l'unanimité du type devrait être respectée. Le suivant traduit la notion de monotonie : si un type T_i envie un autre type T_j , et que l'on rajoute à T_j un agent qui possède un objet de très bonne qualité (c'est-à-dire que tous les agents du premier type souhaiteraient obtenir), alors cette envie ne doit pas "disparaître" : le type doit toujours être envieux. Formellement :

Axiome 3.3.4: Monotonie par ajout d'un objet unanimement envié

Soit une instance I contenant l'ensemble \mathcal{N}^I d'agents partitionnés en $k > 2$ types et l'ensemble d'objets \mathcal{M} .

Soit deux types T_p^I et T_q^I de I et une allocation A^I .

Nous construisons l'instance I' à partir de l'instance I en ajoutant un agent a_0 à T_q et un objet h_0 à A_q tel que $\forall a \in T_p, u_a(A^I(a)) < u_a(h_0)$. Nous avons donc $\mathcal{N}^{I'} = \mathcal{N}^I \cup \{a_0\}$, $\mathcal{M}^{I'} = \mathcal{M}^I \cup \{h_0\}$ et $T_q^{I'} = T_q^I \cup \{a_0\}$. Nous considérons l'allocation $A^{I'}$ définie par $\forall p \neq q, A_p^{I'} = A_p^I$ et $A_q^{I'} = A_q^I \cup \{h_0\}$.

Si T_p^I envie T_q^I dans l'allocation A^I , alors $T_p^{I'}$ envie $T_q^{I'}$ dans l'allocation $A^{I'}$.

Le dernier axiome aborde une notion intéressante : celle de la monotonie. Nous nous contenterons pour l'instant de cet axiome et nous débattons plus en détails de la notion un peu plus loin.

3.3.2 Notions d'envie existantes

Maintenant que nos axiomes ont été posés, nous pouvons vérifier dans quelle mesure les notions d'envie existantes les vérifient. Commençons par lister les définitions d'envie qui s'appliquent naturellement au cas du problème de *house allocation* dans le cadre de type d'agents.

La première est la définition classique, sans modification, simplement appliquée aux types plutôt qu'aux agents, telle qu'utilisée par Benabbou *et al.* [2018] : un type est envieux si il estime le lot d'objets d'un autre agent comme meilleur que le sien, d'après sa fonction d'utilité.

Formellement :

Définition 3.3.1: Sans-envie par type (TEF)

Soit une instance I dans une allocation A .

On dit que le type $T_p \in \mathcal{T}$ envie le type T_q pour l'envie par type si et seulement si $V_p(A_p) < V_q(A_q)$.

On dit que l'allocation A est sans-envie par type (TEF) si et seulement si il n'existe pas de types $T_p \in \mathcal{T}, T_q \in \mathcal{T}$ tels que T_p envie T_q

Cette notion d'envie se comporte de manière très similaire au cadre usuel. La principale différence vient du fait que l'utilité des agents y est habituellement additive, alors qu'ici les types ne peuvent utiliser qu'un certain nombre d'objets au maximum, par conséquent leurs fonctions d'utilité ne sont pas additives. Par exemple, l'utilité d'un lot avec autant d'objets qu'il y a d'agents dans le type peut rester identique lorsqu'on rajoute des objets, même si ces derniers ont des utilités non nulles: il n'est tout simplement pas possible de les allouer à des agents.

Exemple 3.3.1: Exemple d'application de TEF

Considérons une instance I contenant un ensemble d'agents \mathcal{N} partitionnés en deux types $T_1 = \{a_1, a_2\}$ et $T_2 = \{a_3\}$, et un ensemble d'objets $\mathcal{M} = \{h_1, h_2, h_3\}$. Nous posons les fonctions d'utilité des agents :

	h_1	h_2	h_3
a_1	2	1	3
a_2	1	2	3
a_3	4	1	3

Considérons l'allocation A telle que $A_1 = \{h_1, h_2\}$ et $A_2 = \{h_3\}$. Les allocations individuelles seront alors $A(a_1) = h_1$, $A(a_2) = h_2$ et $A(a_3) = h_3$.

Dans cette allocation,

$$V_1(A_1) = 2 + 2 = 4 > V_1(A_2) = 3 + 0$$

donc T_1 n'envie pas T_2 pour TEF.

En revanche,

$$V_2(A_2) = 3 < V_2(A_1) = 4$$

donc T_2 envie T_1 pour TEF.

Une deuxième notion d'envie envisageable considérant des groupes d'agents consiste en une adaptation à notre cadre de la notion d'envie pondérée (Chakraborty *et al.* [2021]), conçue pour un cadre où chaque agent dispose d'un poids qui représente son importance. Cette définition de l'envie pondère les utilités des lots d'objets par le poids des agents qui les possèdent. Cette notion s'adapte naturellement à notre cadre, en considérant simplement les types à la place des agents, et en leur donnant une pondération égale à leur taille.

Définition 3.3.2: Absence d'envie pondérée (WEF)

Soit une instance I dans une allocation A .

On dit que le type $T_p \in \mathcal{T}$ envie le type T_q pour l'envie pondérée si et seulement

$$\text{si } \frac{V_p(A_p)}{|T_p|} < \frac{V_p(A_q)}{|T_q|}.$$

On dit que l'allocation A est sans-envie pondérée (WEF) si et seulement si il n'existe pas de types $T_p \in \mathcal{T}, T_q \in \mathcal{T}$ tels que T_p envie T_q

Lorsque l'on utilise cette notion, les types souhaiteront obtenir des lots d'objets de qualité proportionnelle à leur taille. Par exemple, si un type est deux fois plus grand qu'un autre, il sera envieux s'il n'obtient pas deux fois plus d'utilité avec son lot qu'avec celui de l'autre type.

Exemple 3.3.2: Exemple d'application de WEF

Considérons une instance I contenant un ensemble d'agents \mathcal{N} partitionnés en deux types $T_1 = \{a_1, a_2\}$ et $T_2 = \{a_3\}$, et un ensemble d'objets $\mathcal{M} = \{h_1, h_2, h_3\}$. Nous posons les fonctions d'utilité :

	h_1	h_2	h_3
a_1	2	1	3
a_2	1	2	3
a_3	4	1	3

Considérons l'allocation A telle que $A_1 = \{h_1, h_2\}$ et $A_2 = \{h_3\}$. Les allocations individuelles seront alors $A(a_1) = h_1$, $A(a_2) = h_2$ et $A(a_3) = h_3$.

Dans cette allocation,

$$\frac{V_1(A_1)}{|T_1|} = \frac{2+2}{2} = 2 < \frac{V_1(A_2)}{|T_2|} = \frac{3+0}{1}$$

donc T_1 envie T_2 pour WEF.

En revanche,

$$\frac{V_2(A_2)}{|T_2|} = 3 > \frac{V_2(A_1)}{|T_1|} = \frac{4}{2} = 2$$

donc T_2 n'envie pas T_1 pour WEF.

Une autre notion d'envie entre groupes a été introduite par Aleksandrov et Walsh [2018a]. Plutôt que de considérer l'utilité qu'un lot d'objets peut effectivement apporter à un type, cette notion compare l'utilité des objets du lot moyennée sur tous les agents du type.

Définition 3.3.3: Absence d'envie pour la mesure d'Aleksandrov et Walsh (A&W)

Soit une instance I dans une allocation A .

On dit que le type $T_p \in \mathcal{T}$ envie le type T_q pour la mesure d'Aleksandrov et Walsh si et seulement si :

$$\frac{1}{|T_p|} \sum_{i \in T_p} u_i(A(i)) < \frac{1}{|T_p| \cdot |T_q|} \sum_{i \in T_p} \sum_{h \in A_q} u_i(h)$$

On dit que l'allocation A est sans-envie pour la mesure d'Aleksandrov et Walsh (A&W) si et seulement si il n'existe pas de types $T_p \in \mathcal{T}$, $T_q \in \mathcal{T}$ tels que T_p envie T_q

Notons que cette envie est assez différente des autres notions puisqu'elle considère seulement les utilités des agents, et pas celle du type.

Exemple 3.3.3: Exemple d'application de A&W

Considérons une instance I contenant un ensemble d'agents \mathcal{N} partitionnés en deux types $T_1 = \{a_1, a_2\}$ et $T_2 = \{a_3\}$, et un ensemble d'objets $\mathcal{M} = \{h_1, h_2, h_3\}$. Nous posons les fonctions d'utilité :

	h_1	h_2	h_3
a_1	2	1	3
a_2	1	2	3
a_3	4	1	3

Considérons l'allocation A telle que $A_1 = \{h_1, h_2\}$ et $A_2 = \{h_3\}$. Les allocations individuelles seront alors $A(a_1) = h_1$, $A(a_2) = h_2$ et $A(a_3) = h_3$.

Dans cette allocation,

$$\frac{1}{|T_1|} V_1(A_1) = \frac{1}{2}(2+2) = 2 < \frac{1}{|T_1| \cdot |T_2|} \sum_{i \in T_1} \sum_{h \in A_2} u_i(h) = \frac{1}{2}(3+3) = 3$$

donc T_1 envie T_2 pour A&W.

En revanche,

$$\frac{1}{|T_2|} V_2(A_2) = 3 > \frac{1}{|T_1| \cdot |T_2|} \sum_{i \in T_2} \sum_{h \in A_1} u_i(h) = \frac{1}{2}(4+1) = 2,5$$

donc T_2 n'envie pas T_1 pour A&W.

3.3.3 Vérifications des axiomes

Lorsque l'on vérifie si chaque notion respecte les différents axiomes que nous avons énoncés précédemment, on obtient les résultats suivants :

Notion	axiome 3.3.1	axiome 3.3.2	axiome 3.3.3	axiome 3.3.4
TEF	Oui	Non	Oui	Oui
WEF	Oui	Non	Non	Non
A & W	Non	Oui	Oui	Oui

Nous remarquons qu'aucune notion ne satisfait tous les axiomes, et qu'aucun axiome n'est satisfait par toutes les notions. TEF et A&W semblent être les notions qui vérifient le plus grand nombre d'axiomes, et WEF celle qui en vérifie le moins, malgré son utilisation dans le cadre de fonctions d'utilité additives (Scarlett *et al.* [2023]).

Preuve que TEF vérifie l'axiome 3.3.1. Cet axiome est immédiat par définition. \square

Preuve que TEF ne vérifie pas l'axiome 3.3.2. Considérons une instance I avec $\mathcal{M} = \{h_1, h_2, \dots, h_{103}\}$ objets et deux types d'agents : $T_1 = \{a_1, a_2, \dots, a_{100}\}$ et $T_2 = \{a_{101}, a_{102}, a_{103}\}$. Nous définissons les fonctions d'utilité des agents de T_1 (les utilités des agents de T_2 n'importent pas ici) comme :

$$\forall a \in T_1, \forall j \in [[1, 100]], u_a(h_j) = 1$$

$$\forall a \in T_1, \forall j \in [[101, 103]], u_a(h_j) = 3$$

Nous considérons l'allocation A définie par : $A_1 = \{h_1, h_2, \dots, h_{100}\}$ et $A_2 = \{h_{101}, h_{102}, h_{103}\}$.

Nous avons alors : $\forall a \in T_1, \forall a' \in T_2, u_a(A(a)) = 1 < u_a(A(a')) = 3$, donc l'axiome 3.3.2 impose que T_p envie T_q . Toutefois, comme $V_p(A_p) = 100 > 3 = V_p(A_q)$, l'axiome n'est pas vérifié. \square

Preuve que TEF vérifie l'axiome 3.3.3. Comme l'utilité des types est monotone, nous savons qu'en particulier pour tout type d'agents T_p et tout lot d'objets H_1 et H_2 , nous aurons $V_p(H_1 \cup H_2) \geq \max(H_1, H_2)$. Par conséquent, pour une instance I contenant les types T_p et T_q , si dans une allocation A nous avons $V_p(A_p \cup A_q) = V_p(A_p)$, alors $V_p(A_p) \geq V_p(A_q)$ donc T_p n'envie pas T_q . \square

Preuve que TEF vérifie l'axiome 3.3.4. Si nous avons I et I' tels que décrits dans la définition de l'axiome, alors puisque T_p envie T_q , nous savons que $V_p(A_p) < V_p(A_q)$. Or, le type T_p de I et le type T'_p de I' sont identiques. De plus $A_p = A'_p$, donc $V_p(A_p) = V_p(A'_p)$. En revanche, $V_p(A_q) \leq V_p(A_q \cup \{h_0\})$ par monotonie de la fonction d'utilité. Par conséquent, $V_p(A'_p) < V_p(A_q \cup \{h_0\})$. Donc T'_p envie bien T'_q dans l'allocation A' . \square

Preuve que WEF vérifie l'axiome 3.3.1. De toute évidence, si $|T_p| = |T_q|$, alors $V_p(A_p) < V_p(A_q) \Leftrightarrow \frac{V_p(A_p)}{|T_p|} < \frac{V_p(A_q)}{|T_q|}$ \square

Preuve que WEF ne vérifie pas l'axiome 3.3.2. Considérons l'instance I contenant les types d'agents $T_1 = \{a_1, a_2\}$ et $T_2 = \{a_3, a_4, a_4, a_6\}$, et l'ensemble d'objets $\mathcal{M} = \{h_1, \dots, h_6\}$. Nous posons les fonctions d'utilité des agents a_1 et a_2 telles que:

$$u_{a_1}(h_1) = u_{a_1}(h_2) = u_{a_2}(h_1) = u_{a_2}(h_2) = 6$$

$$\forall i \in \{3, 4, 5, 6\}, u_{a_1}(h_i) = u_{a_2}(h_i) = 10$$

Considérons l'allocation A définie par $A_1 = \{h_1, h_2\}$ et $A_2 = \{h_3, h_4, h_5, h_6\}$. Dans cette allocation, chaque agent de T_1 obtient une utilité de 6 pour son objet et obtiendrait une utilité de 10 de n'importe quel objet de A_2 .

Par conséquent, l'axiome 3.3.2 impose que le type T_1 envie le type T_2 . Cependant,

$$\frac{V_1(A_1)}{|T_1|} = \frac{6 \times 2}{2} = 6 > \frac{V_1(A_2)}{|T_2|} = \frac{10 \times 2}{4} = 5$$

Donc T_1 n'envie pas T_2 : l'axiome n'est pas respecté. \square

Preuve que WEF ne vérifie pas l'axiome 3.3.3. Considérons une instance I constituée de deux types $T_1 = \{a_1, a_2\}$ et $T_2 = \{a_3\}$, et de l'ensemble d'objets $\mathcal{M} = \{h_1, h_2, h_3\}$. Nous définissons les utilités de la manière suivante :

$u_{a_i}(h)$	h_1	h_2	h_3
a_1	10	0	10
a_2	0	1	0
a_3	10	0	10

Si nous nous plaçons dans l'allocation A telle que $A_1 = \{h_1, h_2\}$ et $A_2 = \{h_3\}$, nous avons $V_1(A_1) = V_1(A_1 \cup A_2) = 11$ puisque le type T_1 n'a aucun moyen d'utiliser l'objet h_3 en plus de son lot d'objets A_1 .

Par conséquent l'axiome 2 imposerait que T_1 n'envie pas T_2 . Cependant,

$$\frac{V_1(A_1)}{|T_1|} = \frac{10+1}{2} < \frac{10}{1} = \frac{V_1(A_2)}{|T_2|}$$

Donc T_1 envie T_2 pour l'envie pondérée. Par conséquent, l'envie pondérée ne vérifie pas l'axiome 3.3.3 \square

Preuve que WEF ne vérifie pas l'axiome 3.3.4. Considérons une instance I constituée de deux types $T_1 = \{a_1\}$ et $T_2 = \{a_2, a_3\}$, et de l'ensemble d'objets $\mathcal{M} = \{h_1, h_2, h_3\}$. Nous définissons les utilités de a_1 de la manière suivante : $u_{a_1}(j_1) = 2$ et $u_{a_1}(j_2) = u_{a_1}(j_3) = 5$.

Considérons l'allocation A telle que $A_1 = \{h_1\}$ et $A_2 = \{h_2, h_3\}$. Dans cette allocation, T_1 envie T_2 pour l'envie pondérée puisque

$$\frac{V_1(A_1)}{|T_1|} = \frac{2}{1} < \frac{5}{2} = \frac{V_1(A_2)}{|T_2|}$$

Construisons maintenant l'instance I' en rajoutant l'agent a_4 à T_2 et l'objet h_4 à A_2 . Nous mettons à jour la fonction d'utilité de a_1 en posant $u_{a_1}(h_4) = 5$. Dans cette allocation A' , T_1 n'envie plus T_2 pour l'envie pondérée, puisque

$$\frac{V_1(A_1)}{|T_1|} = \frac{2}{1} > \frac{5}{3} = \frac{V_1(A_2)}{|T_2|}$$

Par conséquent, l'envie pondérée ne vérifie pas l'axiome 3.3.4 □

Preuve que la mesure d'A&W vérifie l'axiome 3.3.2. Soit une instance I contenant deux types T_1 et T_2 et une allocation A telle que:

$$\forall a \in T_p, \forall h \in A_q, u_a(A(a)) < u_a(h)$$

Nous avons alors :

$$\forall a \in T_p, \sum_{h \in A_q} u_a(A(a)) = |T_q| u_a(A(a)) < \sum_{h \in A_q} u_a(h)$$

d'où

$$\sum_{a \in T_p} |T_q| u_a(A(a)) < \sum_{a \in T_p} \sum_{h \in A_q} u_a(h)$$

ce qui nous donne :

$$\frac{1}{|T_p|} \sum_{a \in T_p} u_a(A(a)) < \frac{1}{|T_p| \cdot |T_q|} \sum_{a \in T_p} \sum_{h \in A_q} u_a(h)$$

T_1 enviera donc bien T_2 pour la mesure d'A&W, cette dernière vérifie l'axiome 3.3.2. □

Preuve que la mesure d'A&W vérifie l'axiome 3.3.3. Plutôt que de montrer que la mesure d'A&W vérifie l'axiome 3.3.3 directement, nous utiliserons la définition donnée dans la proposition 3.3.1.

De manière similaire à la preuve pour l'axiome précédent, soit une instance I contenant deux types T_1 et T_2 et une allocation A telle que :

$$\forall a \in T_p, \forall h \in A_q, u_a(A(a)) \geq u_a(h)$$

On a alors:

$$\forall a \in T_p, \sum_{h \in A_q} u_a(A(a)) = |T_q| u_a(A(a)) \geq \sum_{h \in A_q} u_a(h)$$

d'où

$$\sum_{a \in T_p} |T_q| u_a(A(a)) \geq \sum_{a \in T_p} \sum_{h \in A_q} u_a(h)$$

ce qui nous donne :

$$\frac{1}{|T_p|} \sum_{a \in T_p} u_a(A(a)) \geq \frac{1}{|T_p| \cdot |T_q|} \sum_{a \in T_p} \sum_{h \in A_q} u_a(h)$$

□

Preuve que la mesure d'A&W vérifie l'axiome 3.3.4. Si nous avons I et I' tels que décrits dans la définition de l'axiome, alors puisque T_p envie T_q , nous savons que :

$$\frac{1}{|T_p|} \sum_{a \in T_p} u_a(A(a)) < \frac{1}{|T_p| \cdot |T_q|} \sum_{a \in T_p} \sum_{h \in A_q} u_a(h)$$

donc

$$|T_q| \sum_{a \in T_p} u_a(A(a)) < \sum_{a \in T_p} \sum_{h \in A_q} u_a(h)$$

Or par hypothèse $\forall a \in T_p, u_a(A^I(a)) < u_a(h_0)$ donc

$$\sum_{a \in T_p} u_a(A^I(a)) < \sum_{a \in T_p} u_a(h_0)$$

En additionnant ces deux équations, nous obtenons :

$$\left(|T_q| \sum_{a \in T_p} u_a(A(a)) \right) + \sum_{a \in T_p} u_a(A(a)) < \left(\sum_{a \in T_p} \sum_{h \in A_q} u_a(h) \right) + \sum_{a \in T_p} u_a(h_0)$$

D'où :

$$(|T_q| + 1) \sum_{a \in T_p} u_a(A(a)) < \sum_{a \in T_p \cup \{h_0\}} \sum_{h \in A_q} u_a(h)$$

Donc, avec $|T_p| + 1 = |T_p \cup \{h_0\}| = |T'_q|$, nous avons :

$$\frac{1}{|T_p|} \sum_{a \in T_p} u_a(A(a)) < \frac{1}{|T_p| \cdot |T'_q|} \sum_{a \in T_p} \sum_{h \in A'_q} u_a(h)$$

Le type T_p envie donc bien T'_q : la mesure d'Aleksandrov et Walsh vérifie l'axiome 3.3.4.

□

Preuve que la mesure d'A&W ne vérifie pas l'axiome 3.3.1. Considérons une instance I contenant deux types d'agents $T_1 = \{a_1, a_2\}$ et $T_2 = \{a_3, a_4\}$ et un ensemble d'objets $\mathcal{M} = \{h_1, h_2, h_3, h_4\}$. Nous définissons les fonctions d'utilité comme :

$u_{a_i}(h)$	h_1	h_2	h_3	h_4
a_1	5	5	10	0
a_2	5	5	0	10

Nous considérons l'allocation A définie par : $A_1 = \{h_1, h_2\}$ et $A_2 = \{h_3, h_4\}$. Dans cette allocation, $V_1(A_1) = 5 + 5 < V_1(A_2) = 10 + 10$, donc l'axiome 3.3.1 requiert que T_1 envie T_2 .

Or,

$$\frac{1}{|T_1|} \sum_{a \in T_1} u_a(A(a)) = \frac{1}{2}(5 + 5) = 5$$

et

$$\frac{1}{|T_1| \cdot |T_2|} \sum_{a \in T_1} \sum_{h \in A_2} u_a(h) = \frac{1}{4}(10 + 0 + 0 + 10) = 5$$

Donc T_1 n'envie pas T_2 pour la mesure d'A&W : cette dernière ne vérifie pas l'axiome 3.3.1 \square

3.4 Une nouvelle famille de notions d'envie

Les comportements incohérents mis en évidence dans la section précédente ont pour origine les interactions entre l'absence d'additivité de la fonction d'utilité (due au cadre du problème de *house allocation*) et les groupes d'agents de tailles différentes. Pour palier à ces problèmes, nous allons définir une nouvelle famille de notions d'envie, basée sur l'idée de se ramener à des comparaisons entre des groupes de tailles égales en ne considérant que des sous-groupes du plus grand des deux types. Ainsi, lorsque qu'un type T_p se compare à un type T_q :

- si $|T_p| < |T_q|$, alors T_p se comparera à des sous-groupes $G \subseteq T_q$ tels que $|T_p| = |G|$.
- si $|T_p| > |T_q|$, alors les sous-groupes $G \subseteq T_p$ tels que $|G| = |T_q|$ se compareront à T_q .

Nous allons proposer trois notions basées sur ce principe :

- l'envie pour au moins un sous-groupe : T_p envie T_q si au moins une des comparaisons est envieuse.
- l'envie pour tout sous-groupe : T_p envie T_q si toutes les comparaisons sont envieuses.
- l'envie pour k sous-groupes : T_p envie T_q si au moins k comparaisons sont envieuses.

Nous allons étudier chaque notion séparément.

3.4.1 Envie pour au moins un sous-groupe

Définition et axiomes

Des trois notions que nous proposons, l'envie pour au moins un sous-groupe est la plus directe et de la plus facile à calculer. Nous définissons cette notion formellement de la manière suivante :

Définition 3.4.1: Envie pour au moins un sous-groupe

Pour une instance I et une allocation A sur cette instance, nous dirons que le type T_p envie le type T_q si et seulement si au moins l'une des conditions suivantes est vérifiée :

- $|T_p| \leq |T_q|$ et $V_p(A_p) < V_p(A(G))$ pour au moins un sous-groupe $G \subseteq T_q$ tel que $|G| = |T_p|$.
- $|T_p| \geq |T_q|$ et $V_G(A(G)) < V_G(A_q)$ pour au moins un sous-groupe $G \subset T_p$ tel que $|G| = |T_q|$.

Notons que si nous avons $|T_p| = |T_q|$, alors les deux conditions sont toutes les deux vérifiées et les sous-groupes sont les types tout entier.

Le concept de l'envie pour au moins un sous-groupe consiste à ramener la comparaison à des types de même taille en considérant les sous-groupes du plus grand des deux types. Ainsi, si le type T_p souhaite se comparer à T_q , si il est plus petit il cherchera un sous-groupe de T_q de même taille que lui-même qu'il envie pour l'envie classique; si il possède la plus grande taille il cherchera si il contient un sous-groupe de même taille que T_q qui envie T_q pour l'envie classique.

Exemple 3.4.1: Exemple d'utilisation de l'envie pour au moins un sous-groupe

Considérons une instance I contenant un ensemble d'agents \mathcal{N} partitionnés en deux types $T_1 = \{a_1, a_2\}$ et $T_2 = \{a_3, a_4, a_5\}$, et un ensemble d'objets $\mathcal{M} = \{h_1, h_2, h_3, h_4, h_5\}$. Nous posons les fonctions d'utilité :

	h_1	h_2	h_3	h_4	h_5
a_1	1	1	1	1	0
a_2	0	1	0	1	0
a_3	0	1	1	0	0
a_4	0	1	0	0	0
a_5	1	1	0	0	1

Considérons l'allocation A telle que $A_1 = \{h_1, h_2\}$ et $A_2 = \{h_3, h_4, h_5\}$. Les allocations individuelles seront alors $A(a_1) = h_1$, $A(a_2) = h_2$, $A(a_3) = h_3$, $A(a_4) = h_4$ et $A(a_5) = h_5$.

Dans cet exemple, T_1 va se comparer à chaque sous-groupe G de 2 agents de T_2 :

- pour $G = \{a_3, a_4\}$, $V_1(A_1) = 2 = V_1(A(G)) = 2$: pas d'envie.
- pour $G = \{a_3, a_5\}$, $V_1(A_1) = 2 > V_1(A(G)) = 1$: pas d'envie.
- pour $G = \{a_4, a_5\}$, $V_1(A_1) = 2 > V_1(A(G)) = 1$: pas d'envie.

Donc T_1 n'envie pas T_2 pour l'envie pour au moins un sous-groupe

De l'autre coté, chaque sous-groupe G de 2 agents de T_2 va se comparer à T_1 :

- pour $G = \{a_3, a_4\}$, $V_G(A_G) = 1 = V_G(A_1) = 1$: pas d'envie.
- pour $G = \{a_3, a_5\}$, $V_G(A_G) = 2 = V_G(A_1) = 2$: pas d'envie.
- pour $G = \{a_4, a_5\}$, $V_G(A_G) = 1 < V_G(A_1) = 2$: G envie T_1 .

Donc T_2 envie T_1 pour l'envie pour au moins un sous-groupe

Cette notion d'envie vérifie tous les axiomes précédemment énoncés :

Proposition 3.4.1: Vérification des axiomes

l'envie pour au moins un sous-groupe vérifie tous les axiomes présentés dans la partie 3.3.1 :

- l'envie pour au moins un sous-groupe est une extension de la notion d'envie classique (axiome 3.3.1)
- l'envie pour au moins un sous-groupe respecte l'envie par consensus (axiome 3.3.2)
- l'envie pour au moins un sous-groupe respecte la non-envie par indifférence (axiome 3.3.3)
- l'envie pour au moins un sous-groupe est monotone par ajout d'objet unanimement envié (axiome 3.3.4)

Preuve de vérification de l'axiome 3.3.1. Trivial puisque par définition si $|T_q| = |T_p|$ alors G doit être égal soit à T_p soit à T_q . T_p enviera donc T_q si et seulement si $V_p(A_p) < V_p(A_q)$ \square

Preuve de vérification de l'axiome 3.3.2. Considérons une instance I contenant deux types d'agents T_p et T_q dans une allocation A telle que $\forall a \in T_p, \forall h \in A_q, u_a(A(a)) < u_a(h)$. Il est évident que toute allocation des objets de A_q aux agents de T_p donnera une meilleure utilité aux agents impliqués, c'est-à-dire que si l'on considère un type d'agents $G_a \in$

T_p et un groupe d'objets $G_h \in A_q$ tels que $|G_a| = |G_h|$, alors pour toute allocation A_G des objets de G_h aux agents de G_a , nous avons $\forall a \in G_a, u_a(A(a)) < u_a(A_G(a))$ et donc $\sum_{a \in G_a} u_a(A(a)) < \sum_{a \in G_a} u_a(A_G(a))$. C'est en particulier vrai pour A_G^* l'allocation qui maximise $\sum_{a \in G_a} u_a(A_G(a))$ (auquel cas les sommes seront égales à $V_{G_a}(A(G_a))$ et $V_{G_a}(G_h)$ respectivement), et c'est aussi en particulier vrai si $G_a = T_p$ ou $G_h = A_q$. Par conséquent, nous avons, en fonction des tailles des types :

- si $|T_p| \geq |T_q|$: alors pour tout $G \subseteq T_p$ tel que $|G| = |T_q|$, $V_G(A(G)) < V_G(A_q)$
- si $|T_p| \leq |T_q|$: alors pour tout $G \subseteq T_q$ tel que $|G| = |T_p|$, $V_p(A_p) < V_p(A(G))$

Dans les deux cas, T_p enviera T_q pour l'envie pour au moins un sous-groupe. □

Preuve de vérification de l'axiome 3.3.3. Considérons une instance I contenant deux types d'agents T_p et T_q dans une allocation A telle que $V_p(A_p \cup A_q) = V_p(A_p)$. Nous allons raisonner par l'absurde et supposer que T_p envie T_q pour atteindre une contradiction. Cette contradiction viendra du fait que la présence d'envie implique qu'il existe au moins un objet "intéressant" dans A_q , qui permettrait à T_p d'augmenter son utilité. Cependant, nous devons considérer séparément deux cas :

- si $|T_p| \leq |T_q|$: il existe $G \subseteq T_q$ tel que $|G| = |T_p|$ tel que $V_p(A_p) < V_p(A_q)$. Par conséquent, il existe (au moins) un agent $a \in A_p$ et un objet $h \in A(G)$ tels que $u_a(A(a)) < u_a(h)$. Cela veut dire que $V_p(A_p \cup \{h\}) \geq V_p(A_p) - u_a(A(a)) + u_a(h) > V_p(A_p)$, puisque $V_p(A_p - u_a(A(a)) + u_a(h))$ est l'utilité que T_p obtient de l'allocation optimale de A_p à ses agents à laquelle on a remplacé $A(a)$ par h . Nous avons donc $V_p(A_p \cup A_q) \geq V_p(A_p \cup \{h\}) > V_p(A_p)$, ce qui est en contradiction avec les hypothèses.
- si $|T_p| \geq |T_q|$: il existe $G \subseteq T_p$ tel que $|G| = |T_q|$ tel que $V_G(A(G)p) < V_G(A_q)$. Pour affirmer, comme dans le cas précédent, qu'il existe (au moins) un agent $a \in G$ et un objet $h \in A_q$ tels que $u_a(A(a)) < u_a(h)$, il nous faut de plus préciser que si $a \in G$, alors si A^* est l'allocation des objets de $A(G)$ aux agents de G qui maximise la valeur $\sum_{a \in G} u_a(A^*(a))$, nous avons $\forall a \in G, A(a) = A^*(a)$ (c'est-à-dire que cette allocation est identique à l'allocation A sur l'ensemble G). Le reste de la preuve est similaire au cas précédent : nous obtenons $V_p(A_p \cup \{h\}) \geq V_p(A_p) - u_a(A(a)) + u_a(h) > V_p(A_p)$, puisque $V_p(A_p) - u_a(A(a)) + u_a(h)$ est l'utilité que T_p obtient de l'allocation optimale de A_q à ses agents dans laquelle on a remplacé $A(a)$ par h . Nous avons donc $V_p(A_p \cup A_q) \geq V_p(A_p \cup \{h\}) > V_p(A_p)$, ce qui est une contradiction avec les hypothèses.

□

Preuve de vérification de l'axiome 3.3.4. Considérons les instances I et I' , les types d'agents T_p et T_q , un agent a_0 et un objet h_0 tels que décrits dans la définition de l'axiome 3.3.4. Notre preuve doit considérer deux cas différents, en fonction des tailles des types T_p et T_q :

- si $|T_p| \leq |T_q|$: puisque T_p envie T_q , nous savons qu'il existe $G \subseteq T_q$ avec $|G| = |T_p|$, tel que $V_p(A_p) < V_p(A(G))$. Comme en particulier $G \subset T_q \cup \{a_0\}$, nous pouvons considérer le même sous-groupe G pour affirmer que T_p envie $T_q \cup \{a_0\}$ dans I' .
- si $|T_p| > |T_q|$: puisque T_p envie T_q , nous savons qu'il existe $G \subseteq T_p$ avec $|G| = |T_q|$, tel que $V_G(A(G)) < V_G(A_q)$. Considérons un agent quelconque a' de $T_p \setminus G$, et l'ensemble $G' = G \cup \{a'\}$. Nous savons que $u_{a'}(A(a')) < u_{a'}(h_0)$ et que $V_{G'}(A(G')) = V_G(A(G)) + u_{a'}(A(a'))$. De plus, la définition de la fonction V implique que $V_{G'}(A_q \cup \{h_0\}) \geq V_G(A_q) + u_{a'}(A(a'))$ (puisque $V_{G'}$ calcule l'utilité d'une allocation optimale, donc d'utilité au moins aussi grande que celles des allocations où a' reçoit h_0). Par conséquent, $V_{G'}(A_q \cup \{h_0\}) \geq V_G(A_q) + u_{a'}(A(a')) > V_G(A(G)) + u_{a'} = V_{G'}(A(G'))$. Nous avons donc $G' \subseteq T_p$ tel que $|G'| = |A_q \cup \{h_0\}|$ et $V_{G'}(A(G')) < V_{G'}(A_q \cup \{h_0\})$. Donc T_p envie $T_q \cup \{a_0\}$.

□

Existence et calcul d'allocation sans envie

Maintenant que nous disposons d'une notion d'envie entre types qui vérifient les axiomes énoncés précédemment, nous pouvons considérer deux problèmes relatifs à l'utilisation de cette notion : existe-t-il toujours une allocation sans-envie ? Et si non, est-il difficile de déterminer s'il en existe une ? Malheureusement, comme nous allons le voir, il n'est pas garanti qu'une allocation sans-envie existe, et déterminer si une telle allocation existe est NP-complet.

Proposition 3.4.2: Existence d'allocation sans-envie

Il existe des instances de notre problème pour lesquelles aucune allocation sans-envie n'existe.

Proof. Nous allons prouver ce résultat en construisant un exemple d'instance dans laquelle il n'existe pas d'allocation sans envie. Soit une instance I contenant deux types d'agents $T_1 = \{a_1, \dots, a_6\}$ et $T_2 = \{a_7, a_8, a_9\}$, et un ensemble d'objets \mathcal{M} tel que $|\mathcal{M}| = 4$. Nous définissons les utilités telles que $\forall a \in \mathcal{N}, \forall h \in \mathcal{M}, u_a(h) = 1$.

De toute évidence, dans toute allocation A telle que $|A_2| \geq 1$, au moins 3 agents de T_1 n'obtiendront aucun objet. On aura donc un sous-groupe $G \subset T_1$ constitué de ces 3 agents de taille $|T_2| = 3$ tel que $V_G(A(G)) < V_G(A_2)$. T_1 va donc envier T_2 . En revanche, si A est telle que $|A_2| = 0$, alors pour tout sous-groupe $G \subset T_1$ tel que $|G| = |T_2| = 3$, nous aurons $V_2(A_2) < V_2(A(G))$, donc T_2 enviera T_1 .

Par conséquent, pour cette instance, il n'existe pas d'allocation sans envie. □

Avant de nous intéresser à la détermination de l'existence d'une allocation sans-envie, nous allons prouver la complexité de vérification de l'envie sur une allocation.

Proposition 3.4.3: Vérification du caractère sans-envie d'une allocation

Pour toute instance I , pour toute allocation A sur cette instance, il est possible de vérifier en temps polynomial si cette allocation est sans envie pour l'envie pour au moins un sous-groupe.

Preuve de la propriété 3.4.3. Considérons une instance I quelconque et une allocation A sur I quelconque. En premier lieu, notons que, si pour tout couple de types d'agents de I T_p et T_q nous pouvons tester l'envie en temps polynomial, alors il est possible de vérifier le caractère sans-envie de l'allocation A en faisant un nombre quadratique de ces tests. Par conséquent, il nous suffit de montrer que vérifier l'absence d'envie entre deux types quelconques T_p et T_q est faisable en temps polynomial pour prouver la propriété 3.4.3.

Nous supposons sans restriction de généralité que $|T_p| < |T_q|$ (si $|T_p| = |T_q|$, le calcul de l'envie se ramène au calcul de l'envie classique, qui est polynomial).

- Pour déterminer si T_p envie T_q , il nous faut vérifier si il existe un sous-groupe $G \subset T_q$ tel que $|G| = |T_p|$ et $V_p(A_p) < V_p(A(G))$. Pour cela, il nous suffit de calculer les valeurs de $V_p(A_p)$ et de $V_p(A(G^*))$ où G^* est le "meilleur" sous-groupe c'est-à-dire $G^* = \operatorname{argmax}_{G \subset T_q: |G|=|T_p|} V_p(A(G))$. Or, ces calculs se ramènent au problème du calcul d'une allocation de poids maximum classique, qui est connu pour être polynomial.
- Pour déterminer si T_q envie T_p , de manière similaire, il nous faut vérifier si il existe un sous-groupe $G \subset T_q$ tel que $|G| = |T_p|$ et $V_G(A_p) > V_G(A(G))$. Pour cela, il nous suffit de calculer si $\max_{G \subset T_p: |G|=|T_q|} \{V_G(A_q) - V_G(A(G))\} > 0$, ce qui peut être réduit à un problème d'allocation de poids maximum appliqué à un graphe bipartite $g = (T_p \cup T_q, T_p \times T_q)$ dont les poids sur les arêtes sont définis par $\{u_i(A(i')) - u_i(A(i))\}_{(i,i') \in T_p \times T_q}$. Ce problème classique est solvable en temps polynomial.

On peut donc vérifier l'absence d'envie entre deux types en temps polynomial, c'est donc aussi le cas pour le caractère sans-envie de l'allocation. \square

Nous pouvons maintenant prouver le résultat suivant:

Proposition 3.4.4: Détermination de l'existence d'une allocation sans-envie

La détermination de l'existence d'une allocation sans-envie pour l'envie pour au moins un sous-groupe est un problème NP-complet, même quand les types sont de même taille et que les agents partagent la même utilité.

Preuve de la propriété 3.4.4. Cette preuve est en deux parties. Tout d'abord, nous devons montrer que ce problème est dans NP, puis nous montrerons qu'il est NP-difficile.

Puisque nous pouvons tester si une allocation est sans-envie en temps polynomial, le problème de déterminer si une allocation sans-envie existe pour une instance donnée

peut être résolu par un algorithme non-déterministe qui crée une allocation, vérifie si elle est sans envie (en temps polynomial d'après la proposition 3.4.3), et retourne vrai si c'est le cas et faux sinon. Il est donc dans NP.

Pour montrer que le problème de la détermination de l'existence d'une allocation sans-envie est NP-difficile, nous allons montrer que l'on peut réduire une instance du problème de partition à une instance de ce problème.

Soit une instance du problème de partition définie par l'ensemble d'entiers $S = \{s_1, \dots, s_f\}$ (avec $f \in \mathbb{N}^*$, $\forall i \in \llbracket 1, f \rrbracket, s_i \in \mathbb{N}^*$). Nous construisons une instance de notre problème en définissant l'ensemble des agents \mathcal{N} composé de deux types d'agents T_1 et T_2 tels que $|T_1| = |T_2| = f$, et en définissant l'ensemble des objets \mathcal{M} et les utilités de ces objets de la manière suivante : $\forall i \in \llbracket 1, f \rrbracket$. Nous ajoutons à \mathcal{M} l'objet h_i conférant à tout agent $a \in \mathcal{N}$ l'utilité $u_a(h_i) = s_i$. Puisque les utilités sont identiques pour tous les agents, les fonctions d'utilité n'ont pas besoin de dépendre des agents, nous posons donc : $\forall i \in \llbracket 1, f \rrbracket, u(h_i) = s_i$.

Considérons une allocation A de cette instance. Comme $|T_1| = |T_2| = f$, nous avons nécessairement $|A_1| \leq f$ et $|A_2| \leq f$. De plus, A est sans-envie si et seulement si $V_1(A_1) \geq V_1(A_2)$ et $V_2(A_2) \geq V_2(A_1)$, ce qui revient à dire que $\sum_{h \in A_1} u(h) = \sum_{h \in A_2} u(h)$. Par conséquent, si il existe une allocation A sans-envie, alors nous pouvons construire une partition de S en deux sous-ensembles S_1 et S_2 définis par $S_1 = \{s_i | i \text{ tel que } h_i \in A_1\}$ et $S_2 = \{s_i | i \text{ tel que } h_i \in A_2\}$ qui vérifient $\sum_{s_i \in S_1} s_i = \sum_{h \in A_1} u(h) = \sum_{h \in A_2} u(h) = \sum_{s_i \in S_2} s_i$. La partition $S = S_1 \cup S_2$ est donc une solution au problème de partition. Réciproquement, si il existe une partition de S en deux sous-ensembles S_1 et S_2 tels que $\sum_{s_i \in S_1} s_i = \sum_{s_i \in S_2} s_i$, alors on peut créer l'allocation A telle que $A_1 = \{h_i | i \text{ tel que } s_i \in S_1\}$ et $A_2 = \{h_i | i \text{ tel que } s_i \in S_2\}$ qui sera sans-envie.

Par conséquent, il existe une solution au problème de partition si et seulement si il existe une allocation sans-envie. Comme le problème de partition est NP-difficile, alors le problème de la détermination de l'existence d'une allocation sans-envie l'est aussi.

Ce problème est donc NP-complet. \square

3.4.2 Envie pour tout sous-groupe

À l'opposé de l'envie pour au moins un sous-groupe, l'envie pour tout sous-groupe autorisera un type à être envieux que si les comparaisons avec tous les sous-groupes génèrent de l'envie.

Définition et calcul

Nous définissons cette notion de manière formelle en posant :

Définition 3.4.2: Envie pour tout sous-groupe

Pour une instance I et une allocation A sur cette instance, nous dirons que le type T_p envie le type T_q si et seulement si au moins l'une des conditions suivantes est vérifiée :

- $|T_p| \leq |T_q|$ et $V_p(A_p) < V_p(A(G))$ pour tout sous-groupe $G \subseteq T_q$ tel que $|G| = |T_p|$.
- $|T_p| \geq |T_q|$ et $V_G(A(G)) < V_G(A_q)$ pour tout sous-groupe $G \subset T_p$ tel que $|G| = |T_q|$.

Exemple 3.4.2: Exemple d'utilisation de l'envie pour tout sous-groupe

Considérons une instance I contenant un ensemble d'agents \mathcal{N} partitionnés en deux types $T_1 = \{a_1, a_2\}$ et $T_2 = \{a_3, a_4, a_5\}$, et un ensemble d'objets $\mathcal{M} = \{h_1, h_2, h_3, h_4, h_5\}$. Nous considérons les fonctions d'utilité :

	h_1	h_2	h_3	h_4	h_5
a_1	1	1	1	1	0
a_2	0	0	0	1	0
a_3	0	1	1	0	0
a_4	1	0	0	0	0
a_5	1	0	0	0	0

Considérons l'allocation A telle que $A_1 = \{h_1, h_2\}$ et $A_2 = \{h_3, h_4, h_5\}$. Les allocations individuelles seront alors $A(a_1) = h_1$, $A(a_2) = h_2$, $A(a_3) = h_3$, $A(a_4) = h_4$ et $A(a_5) = h_5$.

Dans cet exemple, T_1 va se comparer à chaque sous-groupe G de 2 agents de T_2 :

- pour $G = \{a_3, a_4\}$, $V_1(A_1) = 1 < V_1(A(G)) = 2$: T_1 envie G .
- pour $G = \{a_3, a_5\}$, $V_1(A_1) = 1 = V_1(A(G)) = 1$: pas d'envie.
- pour $G = \{a_4, a_5\}$, $V_1(A_1) = 1 = V_1(A(G)) = 1$: pas d'envie.

Donc T_1 n'envie pas T_2 pour l'envie pour tout sous-groupe.

De l'autre côté, chaque sous-groupe G de 2 agents de T_2 va se comparer à T_1 :

- pour $G = \{a_3, a_4\}$, $V_G(A_G) = 1 < V_G(A_1) = 2$: G envie T_1 .
- pour $G = \{a_3, a_5\}$, $V_G(A_G) = 1 < V_G(A_1) = 2$: G envie T_1 .
- pour $G = \{a_4, a_5\}$, $V_G(A_G) = 0 < V_G(A_1) = 1$: G envie T_1 .

Donc T_2 envie T_1 pour l'envie pour tout sous-groupe.

Calculer si un type T_p envie un type T_q pour l'envie pour tout sous-groupe peut se ramener à la résolution d'un problème d'optimisation quadratique non convexe.

Nous nous limiterons au cas $|T_p| < |T_q|$, puisqu'il est possible de passer d'un cas à l'autre en transposant la matrice d'utilité (c'est-à-dire en posant $u_{i'}(A(j')) = u_j(A(i))$).

Vérifier si T_p envie T_q pour l'envie pour tout sous-groupe revient à vérifier l'utilité du lot d'objets $V_p(A(G_\perp))$ où $G_\perp = \operatorname{argmin}_{G \subseteq T_q, |G|=|T_p|} V_p(G)$. En effet, puisque tous les autres sous-groupes génèrent une utilité supérieure ou égale, alors si $V_p(T_p) < V(G_\perp)$, on a que pour tout $G \subseteq T_q, |G|=|T_p|$, $V_p(T_p) < V(G)$ (et si $V_p(T_p) > V(G_\perp)$, alors il n'y a pas d'envie pour l'envie pour tout sous-groupe).

Nous souhaitons donc calculer la valeur $V_p(A(G_\perp)) = \min_{G \subseteq T_q, |G|=|T_p|} V_p(G)$. Or nous rappelons que l'utilité $V_p(B)$ d'un lot d'objets B s'obtient par le programme linéaire en nombres entiers :

$$\max \sum_{i=0}^{n-1} \sum_{j=0}^{n-1} x_{i,j} \cdot u_{i,j} \quad (3.1)$$

$$\text{s.t. } \sum_{i=0}^{n-1} x_{i,j} \leq 1 \quad \forall 0 \leq j < n \quad (3.2)$$

$$\sum_{j=0}^{n-1} x_{i,j} \leq 1 \quad 0 \leq i < n \quad (3.3)$$

$$\forall 0 \leq i < n, 0 \leq j < n, x_{i,j} \in \{0, 1\} \quad (3.4)$$

$$(3.5)$$

où:

- $n = |T_p|$
- $u_{i,j} = u_i(A(j))$
- $x_{i,j}$ vaut 1 si et seulement si l'objet $A(j)$ est alloué à l'agent i , et 0 sinon.
- La contrainte 3.2 représente le fait qu'un objet ne peut être alloué qu'à un seul agent.
- La contrainte 3.3 représente le fait qu'un agent ne peut recevoir qu'un seul objet.

Nous allons modifier la forme de ce programme linéaire pour qu'il soit plus adapté à la suite. Nous posons:

- $m = |T_q|$
- $U = \{U_k\}_{k \in \llbracket 0, n \cdot m - 1 \rrbracket}$ tel que $U_{n*j+i} = u_i(A(j))$
- $X = \{X_k\}_{k \in \llbracket 0, n \cdot m - 1 \rrbracket}$ tel que $X_{n*j+i} = x_{i,j}$

- $B = \{b_j\}_{j \in \llbracket 0, m-1 \rrbracket}$ la matrice colonne représentant le sous-groupe G , tel que $b_j = 1$ si et seulement si $j \in G$ et 0 sinon.

Le programme reformulé est donc :

$$\max \sum_{k=0}^{n \cdot m - 1} X_k \cdot U_k \quad (3.6)$$

$$\text{s.t.} \sum_{i=0}^{n-1} X_{j \cdot n + i} \leq b_j \quad \forall 0 \leq j \leq m \quad (3.7)$$

$$\sum_{j=1}^{m-1} X_{j \cdot n + i} \leq 1 \quad \forall 0 \leq i \leq n \quad (3.8)$$

$$\forall 0 \leq k < n \cdot m, X_k \in \{0, 1\} \quad (3.9)$$

$$(3.10)$$

Nous allons maintenant écrire un programme qui détermine le sous-groupe G (via la matrice colonne associée B) qui minimise cette valeur :

$$\min_B \max_X \sum_{k=1}^{n \cdot m} X_k \cdot U_k \quad (3.11)$$

$$\text{s.t.} \sum_{i=0}^{n-1} X_{j \cdot n + i} \leq b_j \quad \forall 0 \leq j \leq m \quad (3.12)$$

$$\sum_{j=1}^{m-1} X_{j \cdot n + i} \leq 1 \quad \forall 0 \leq i \leq n \quad (3.13)$$

$$\sum_{j=1}^{m-1} b_j \leq n \quad (3.14)$$

$$\forall 0 \leq k < n \cdot m, X_k \in \{0, 1\} \quad (3.15)$$

$$\forall 0 \leq j < m, b_j \in \{0, 1\} \quad (3.16)$$

$$(3.17)$$

Cependant, ceci n'est pas un programme linéaire, nous allons donc devoir le modifier pour définir un programme linéaire équivalent.

Revenons au calcul de V_p et remarquons qu'on peut relaxer la contrainte entière pour passer en continu. Concrètement, le programme linéaire en nombres entiers calculant V_p pour un sous-groupe G encodé par la matrice B présentée plus haut, est équivalent au programme linéaire suivant :

$$\max \sum_{k=0}^{n.m-1} X_k \cdot U_k \quad (3.18)$$

$$\text{s.t.} \sum_{i=0}^{n-1} x_{j*n+i} \leq b_j \quad \forall 0 \leq j \leq m \quad (3.19)$$

$$\sum_{j=1}^{m-1} x_{j*n+i} \leq 1 \quad \forall 0 \leq i \leq n \quad (3.20)$$

$$\forall 0 \leq k < n * m, X_k \geq 0 \quad (3.21)$$

$$(3.22)$$

Ce programme linéaire peut admettre des solutions X supplémentaires, mais la valeur du maximum n'augmentera pas.

Puisque la solution optimale est atteinte, nous savons que le programme dual l'atteindra également. Pour calculer le dual, nous allons poser la matrice A telle que les contraintes du programme linéaire soient représentables par $AX \leq B'$ (où B' est la matrice colonne construite à partir de B en rajoutant n éléments de valeur 1 en bas). La matrice A est donc une matrice binaire de taille $(n + m) \times (n \times m)$ telle que : $\forall 0 \leq i < n, 0 \leq j < n, A_{j,j*n+i} = A_{m+i,j*n+i} = 1$ et tous les autres éléments de A soit égaux à zéro.

Le problème dual que l'on obtient est :

$$\min B'^t \cdot Y \quad (3.23)$$

$$\text{s.t.} A^t \cdot Y \geq U \quad (3.24)$$

$$\forall 0 \leq k < n + m, Y_k \geq 0 \quad (3.25)$$

$$(3.26)$$

Précisons qu'étant donné la forme de la matrice A , la contrainte $A^t \cdot Y \geq U$ est équivalente à :

$$\forall 0 \leq i < n, 0 \leq j < m, Y_j + Y_{m+i} \geq U_{j*n+i}$$

Nous pouvons maintenant reprendre la formule complète permettant de calculer l'utilité du lot d'objets du plus petit sous-groupe :

$$\min_{B'} \min_Y \sum_{l=0}^{m+n-1} b'_l \cdot Y_l \quad (3.27)$$

$$\text{s.t. } A^t \cdot Y \geq U \quad (3.28)$$

$$\sum_{j=1}^{m-1} b'_j \leq n \quad (3.29)$$

$$\forall j \geq m, b'_j \geq 1 \quad (3.30)$$

$$\forall 0 \leq l < n+m, Y_l \geq 0 \quad (3.31)$$

$$\forall 0 \leq j < m+n, b'_j \in \{0, 1\} \quad (3.32)$$

$$(3.33)$$

Notons que la contrainte 3.30 sert à forcer les n dernières valeurs de B' à être égales à 1, puisqu'elles ne font pas réellement partie de notre problème.

Nous pouvons facilement faire passer tout le problème en continu en changeant l'ensemble de définition de la ligne 3.32 en $b'_j \geq 0$ et en rajoutant la contrainte : $\forall 0 \leq j < m+n, b'_j \leq 1$. Il est possible que cela augmente l'ensemble des solutions optimales, mais cela ne peut pas augmenter la valeur de l'utilité optimale elle-même.

Il ne nous reste alors plus qu'à fusionner les deux problèmes de minimisation en considérant une matrice colonne qui contiendrait la concaténation de Y et de B' . Cependant, le problème résultant est un problème d'optimisation quadratique non convexe, il n'est donc pas possible de le résoudre en temps polynomial.

Axiomes

Proposition 3.4.5

L'envie pour tout sous groupe vérifie les axiomes 3.3.1, 3.3.2 et 3.3.3, mais pas l'axiome 3.3.4.

Les preuves de vérifications des trois premiers axiomes par l'envie pour tout sous-groupe seront quasiment identiques aux preuves correspondantes de l'envie pour au moins un sous-groupe.

Preuve de vérification de l'axiome 3.3.1. La preuve est triviale puisque par définition si $|T_q| = |T_p|$ alors G doit être égal soit à T_p soit à T_q , il existe donc un seul sous-groupe à considérer. T_p enviera donc T_q si et seulement si $V_p(A_p) < V_p(A_q)$. \square

Preuve de vérification de l'axiome 3.3.2. Considérons une instance I contenant deux types d'agents T_p et T_q dans une allocation A telle que $\forall a \in T_p, \forall h \in A_q, u_a(A(a)) < u_a(h)$. Il est évident que toute allocation des objets de A_q aux agents de T_p donnera une meilleure utilité aux agents impliqués, c'est-à-dire que si l'on considère un type d'agent

$G_a \in T_p$ et un groupe d'objets $G_h \in A_q$ tels que $|G_a| = |G_h|$, alors pour toute allocation A_G des objets de G_h aux agents de G_a , nous avons $\forall a \in G_a, u_a(A(a)) < u_a(A_G(a))$ et donc $\sum_{a \in G_a} u_a(A(a)) < \sum_{a \in G_a} u_a(A_G(a))$. C'est donc en particulier vrai pour A_G^* l'allocation qui maximise $\sum_{a \in G_a} u_a(A_G(a))$ (auxquels cas les sommes seront égales à $V_{G_a}(A(G_a))$ et $V_{G_a}(G_h)$ respectivement), et c'est aussi en particulier vrai si $G_a = T_p$ ou $G_h = A_q$. Par conséquent, nous avons, en fonction des tailles des types :

- si $|T_p| \geq |T_q|$: alors pour tout $G \subseteq T_p$ tel que $|G| = |T_q|$, $V_G(A(G)) < V_G(A_q)$
- si $|T_p| \leq |T_q|$: alors pour tout $G \subseteq T_q$ tel que $|G| = |T_p|$, $V_p(A_p) < V_p(A(G))$

Dans les deux cas, T_p enviera T_q pour l'envie pour tout sous-groupe. \square

Preuve de vérification de l'axiome 3.3.3. Considérons une instance I contenant deux types d'agents T_p et T_q et une allocation A sur I telle que $V_p(A_p \cup A_q) = V_p(A_p)$. Nous allons raisonner par l'absurde et supposer que T_p envie T_q pour atteindre une contradiction. Cette contradiction viendra du fait que la présence d'envie implique qu'il existe au moins un objet "intéressant" dans A_q qui permettrait à T_p d'augmenter son utilité. Cependant, nous devons considérer séparément deux cas :

- si $|T_p| \leq |T_q|$: pour tout $G \subseteq T_q$ tel que $|G| = |T_p|$, nous avons $V_p(A_p) < V_p(A_q)$. Par conséquent, il existe (au moins) un agent $a \in A_p$ et un objet $h \in A(G)$ tels que $u_a(A(a)) < u_a(h)$ (il en existe en fait au moins $|T_q| - |T_p|$, mais un seul nous suffit). Cela veut dire que $V_p(A_p \cup \{h\}) \geq V_p(A_p) - u_a(A(a)) + u_a(h) > V_p(A_p)$, puisque $V_p(A_p - u_a(A(a)) + u_a(h))$ est l'utilité que T_p obtient de l'allocation optimale de A_p à ses agents dans laquelle on a remplacé $A(a)$ par h . Nous avons donc $V_p(A_p \cup A_q) \geq V_p(A_p \cup \{h\}) > V_p(A_p)$, ce qui est en contradiction avec les hypothèses.
- si $|T_p| \geq |T_q|$: pour tout $G \subseteq T_p$ tel que $|G| = |T_q|$ nous avons $V_G(A(G)) < V_G(A_q)$. Pour affirmer, comme dans le cas précédent, qu'il existe (au moins) un agent $a \in T_p$ et un objet $h \in A_q$ tels que $u_a(A(a)) < u_a(h)$, il nous faut de plus préciser que si $a \in G$ pour un $G \subseteq T_p$ tel que $|G| = |T_q|$ quelconque, alors si A^* est l'allocation des objets de $A(G)$ aux agents de G qui maximise la valeur $\sum_{a \in G} u_a(A^*(a))$, nous avons $\forall a \in G, A(a) = A^*(a)$ (c'est-à-dire que cette allocation est identique à l'allocation A sur l'ensemble G). À partir de là, le reste de la preuve est similaire au cas précédent : nous obtenons $V_p(A_p \cup \{h\}) \geq V_p(A_p) - u_a(A(a)) + u_a(h) > V_p(A_p)$, puisque $V_p(A_p) - u_a(A(a)) + u_a(h)$ est l'utilité que T_p obtient de l'allocation optimale de A_q à ses agents à laquelle on a remplacé $A(a)$ par h . Nous avons donc $V_p(A_p \cup A_q) \geq V_p(A_p \cup \{h\}) > V_p(A_p)$, ce qui est une contradiction avec les hypothèses. \square

Preuve de non-vérification de l'axiome 3.3.4. Nous allons présenter un contre-exemple : plaçons nous dans le cas où tous les agents partagent la même fonction d'utilité. Posons $T_p = \{a_1, a_2\}$, $T_q = \{a_3, a_4\}$ et $\mathcal{M} = \{h_1, h_2, h_3, h_4\}$ tels que $u(h_1) = 4$, $u(h_2) = 4$, $u(h_3) = 0$ et $u(h_4) = 10$. Considérons l'allocation $A(a_i) = h_i$ pour tout i . Dans cette allocation, T_p

envie T_q (immédiat par l'axiome 3.3.1). Ajoutons maintenant à T_q l'agent a_5 possédant l'objet h_5 tel que $u(h_5) = 6$. Cet objet est meilleur que h_1 et h_2 , l'axiome imposerait donc à T_p de rester envieux. Cependant, le lot d'objets du sous-groupe $\{a_3, a_5\}$ confère à T_p une utilité de $V_p(\{h_3, h_5\}) = 6 + 0 < V_p(A_p) = 8$, ce qui veut dire que T_p n'est pas envieux pour l'envie pour tout sous-groupe. \square

3.4.3 Envie pour k sous-groupes

Un grand nombre de nuances existent entre l'envie pour au moins un sous-groupe et l'envie pour k sous-groupes : on peut imaginer demander un certain nombre de comparaisons envieuses, ou une certaine proportion. Nous regrouper toutes ces nuances dans l'envie pour k sous-groupes qui permet d'étudier les propriétés communes à toutes les notions créées sur cette base.

Définition et calcul

Définition 3.4.3: Envie pour k sous-groupes

Pour une instance I et une allocation A sur cette instance, nous dirons que le type T_p envie le type T_q si et seulement si au moins l'une des conditions suivantes est vérifiée :

- $|T_p| \leq |T_q|$ et $V_p(A_p) < V_p(A(G))$ pour au moins k sous-groupes différents $G \subseteq T_q$ tel que $|G| = |T_p|$.
- $|T_p| \geq |T_q|$ et $V_G(A(G)) < V_G(A_q)$ pour au moins k sous-groupes différents $G \subset T_p$ tel que $|G| = |T_q|$.

Il est possible de déterminer si un type en envie un autre pour l'envie pour k sous-groupes en temps polynomial en k et en la taille de l'entrée. Cela veut dire que pour un k fixé ne dépendant pas de la taille des groupes, cet algorithme sera polynomial en la taille de l'entrée. Cependant, il ne faut pas oublier que le nombre total de sous-groupes est exponentiel en la taille de l'entrée (il existe $\binom{m}{n}$ sous-groupes où $m = \max(|T_p|, |T_q|)$ et $n = \min(|T_p|, |T_q|)$). Par conséquent, il ne sera pas possible en pratique de garantir que k représente une certaine proportion de sous-groupes sans faire passer l'algorithme de calcul en temps exponentiel.

Comme pour les autres notions, le cas où $|T_p| > |T_q|$ peut se ramener au cas $|T_p| < |T_q|$, nous ne considérerons donc que ce dernier.

L'algorithme vérifiant l'envie pour k sous-groupes est une modification de l'algorithme présenté par Chegiredy et Hamacher [1987] servant à calculer les k meilleurs couplages d'un graphe bipartite. Cet algorithme fonctionne en calculant les k sous-groupes $G \subseteq T_q$ de taille $|T_p|$ dont les lots d'objets ont les plus grandes utilités pour T_p (c'est-à-dire de $V_p(A(G))$ maximum). Cet algorithme se compose de deux parties : une méthode

qui à partir du sous-groupe G de $V_p(A(G))$ maximum calcule le deuxième meilleur groupe, et un algorithme utilisant cette méthode récursivement pour trouver les k meilleurs.

La méthode permettant de calculer la deuxième meilleure allocation utilisera une nouvelle structure que nous appelons "graphe d'allocation" :

Définition 3.4.4: Graphe d'allocation

Pour une instance I contenant les types T_p et T_q tels que $|T_p| < |T_q|$ et un sous-groupe $G \subset T_q$ tel que $|G| = |T_p|$, soit A_G l'allocation optimale des objets de $A(G)$ aux agents de T_p (c'est-à-dire celle utilisée pour calculer $V_p(A(G))$). Nous définissons le **graphe d'allocation** Gr_G comme un graphe bipartite orienté pondéré $Gr_G = (N_G \cup M_G, E_G)$ avec :

- N_G contient un sommet pour chaque agent de T_p
- M_G contient un sommet pour chaque objet de T_q
- $\forall i \in N_G, \forall j \in M_G, (i, j) \in E_G$ si et seulement si $A_G(i) = A(j)$. Son poids $w_{(i,j)}$ est alors défini par $w_{(i,j)} = -u_i(A(j))$
- $\forall i \in N_G, \forall j \in M_G, (j, i) \in E_G$ si et seulement si $A_G(i) \neq A(j)$, et son poids est alors défini par $w_{(j,i)} = u_i(A(j))$.

Ce graphe représente l'allocation des éléments de $A(G)$ aux agents de T_p , le poids de l'arête $(j, i) \in M_G \times N_G$ représente l'utilité que T_p gagnerait à affecter l'objet $A(j)$ à i (en supposant qu'il soit libre), tandis que le poids de l'arête $(i, j) \in (N_G \times M_G) \cap E_G$ représente l'utilité que T_p perdrait si l'agent i perdait l'objet $A(j)$. Ainsi, si l'agent i est affecté à $A(j_1)$ (pour $j_1 \in G$) dans l'allocation A_G , alors si en décidant de l'affecter à $A(j_2)$ ($j_2 \in T_q \setminus G$) l'utilité de la nouvelle allocation sera modifiée de $w_{(j_2,i)} + w_{(i,j_1)}$.

Concrètement, chaque objet pointe vers les agents auxquels il n'est pas alloué avec un poids égal à l'utilité qu'il peut leur apporter, et est pointé par l'agent auquel il est alloué avec un poids égal à l'utilité qu'il peut lui faire perdre. Cette construction va nous permettre de représenter les modifications de l'allocation par des chemins sur le graphe, puisque chaque agent dont l'allocation est modifiée va perdre l'utilité de son objet actuel et gagner celle de son nouvel objet.

À partir de ce graphe, nous allons définir des chemins d'échanges, qui nous serviront à représenter des changements d'allocation :

Définition 3.4.5: Chemin d'échanges et chemin canonique d'échanges

Soit I, T_p, T_q, A, G et Gr_G tels que définis dans la définition 3.4.4. Soit $j_1 \in T_q \setminus G$ et $j_2 \in G$.

$S_{j_1, j_2} \subseteq E_G$ est un **chemin d'échanges** entre j_1 et j_2 si et seulement si S_{j_1, j_2} est un chemin dans Gr_G allant de j_1 à j_2 . On notera $w(S_{j_1, j_2}) = \sum_{e \in S_{j_1, j_2}} w_e$ le poids de S_{j_1, j_2} .

$S_{j_1, j_2} \subseteq E_G$ est un **chemin canonique d'échanges** entre j_1 et j_2 si et seulement si S_{j_1, j_2} est de poids maximum parmi les chemins d'échanges entre j_1 et j_2 .

Remarquons qu'un tel chemin sera alternant entre N_G et M_G , et que tout $(j, i) \in M_G \times N_G$ sera suivi d'un $(i, A(i))$.

Chacun de ces chemins représente une modification de A où chaque agent traversé reçoit l'objet qui le précède. Comme chaque agent a un unique successeur, qui est son ancienne allocation, nous savons qu'il n'y aura pas de conflit. De plus, si le chemin n'est pas une boucle, alors la nouvelle allocation possède un unique objet qui n'était pas présent dans l'ancienne allocation et vice-versa.

La raison pour laquelle nous introduisons la notion de chemin d'échange "canonique" est que cela nous assure que l'allocation résultante soit optimal pour le lot d'objet alloué (prouvé plus loin).

Formellement, nous pouvons définir une nouvelle allocation en retirant $A(i)$ à i et en lui donnant $A(j)$.

Définition 3.4.6: Allocation définie par les chemins d'échanges

Soit I, T_p, T_q, A, G et Gr_G tels que définis dans la définition 3.4.4.

Soit $j_1 \in T_q \setminus G$ et $j_2 \in G$, soit S_{j_1, j_2} un chemin d'échanges entre j_1 et j_2 .

Le chemin S_{j_1, j_2} conduit à une nouvelle allocation $A_{S_{j_1, j_2}}$ des objets de $A((G \setminus \{j_2\}) \cup \{j_1\})$ aux agents de T_p définie par :

$$\forall (j, i) \in S_{j_1, j_2} \cap (M_G \times N_G), A_{S_{j_1, j_2}}(i) = A(j)$$

et $A_{S_{j_1, j_2}}(i) = A_G(i)$ pour les autres i .

Proposition 3.4.6

Soit I, T_p, T_q, A, G et Gr_G tels que définis dans la définition 3.4.4.

Soit $j_1 \in T_q \setminus G$ et $j_2 \in G$, soit S_{j_1, j_2} un chemin d'échanges entre j_1 et j_2 et $A_{S_{j_1, j_2}}$ l'allocation définie par ce chemin.

L'utilité totale de T_p dans l'allocation $A_{S_{j_1, j_2}}$ définie par : $U_p(A_{S_{j_1, j_2}}(T_p)) = \sum_{i \in T_p} u_i(A_{S_{j_1, j_2}}(i))$ est telle que :

$$U_p(A_{S_{j_1, j_2}}(T_p)) = V_p(A_p) + w(S_{j_1, j_2})$$

De plus, si $A_{S_{j_1, j_2}}$ est un chemin d'échanges canonique entre j_1 et j_2 , alors $A_{S_{j_1, j_2}}$ est une allocation optimale des objets de $A((G \setminus \{j_2\}) \cup \{j_1\})$ aux agents de T_p , et donc :

$$U_p(A_{S_{j_1, j_2}}(T_p)) = V_p(A((G \setminus \{j_2\}) \cup \{j_1\}))$$

Preuve. Cette preuve est immédiate par construction :

$$U_p(A_{S_{j_1, j_2}}(T_p)) = V_p(A_p) - \sum_{(i, j) \in S_{j_1, j_2} \cap (N_G \times M_G)} u_i(A(j)) + \sum_{(i, j) \in S_{j_1, j_2} \cap (M_G \times N_G)} u_i(A(j))$$

$$U_p(A_{S_{j_1, j_2}}(T_p)) = V_p(A_p) + \sum_{(i, j) \in S_{j_1, j_2} \cap (N_G \times M_G)} w_{(i, j)} + \sum_{(i, j) \in S_{j_1, j_2} \cap (M_G \times N_G)} w_{(j, i)}$$

donc :

$$U_p(A_{S_{j_1, j_2}}(T_p)) = V_p(A_p) + w(S_{j_1, j_2})$$

En ce qui concerne le cas où S_{j_1, j_2} est un chemin canonique, nous allons devoir montrer que $A_{S_{j_1, j_2}}$ est optimale.

Nous savons déjà que $U_p(A_{S_{j_1, j_2}}(T_p)) \leq V_p(A((G \setminus \{j_2\}) \cup \{j_1\}))$ par définition de V_p .

Posons $G' = G \setminus \{j_2\} \cup \{j_1\}$, et $A_{G'}$ l'allocation optimale des objets de $A(G')$ aux agents de T_p . Nous définissons l'ensemble d'arêtes $S \in E_G$ contenant l'ensemble des arêtes $(A_{G'}(i), i)$ et $(i, A_G(i))$ pour tout agent $i \in T_p$ tel que $A_G(i) \neq A_{G'}(i)$. Comme tous les agents sauf j_1 et j_2 sont présents à la fois dans G et dans G' , ils ne seront soit pas impliqués dans S , soit présents dans deux arêtes (une pour l'agent auquel leur objet est affecté dans A_G , une pour l'agent auquel leur objet est affecté dans $A_{G'}$). Par conséquent, S est un ensemble contenant un chemin d'échanges de j_1 à j_2 et un certain nombre de cycles, tous disjoints (puisque qu'un objet ne peut être alloué qu'à un seul agent et vice-versa).

Cependant, tout cycle $C \in S$ est forcément de poids nul, par l'absurde si ce n'est pas le cas :

- si $w(C) > 0$: à partir de l'allocation A_G , on peut créer une allocation A_{G+C} des objets de $A(G)$ aux agents de T_p en posant $\forall (j, i) \in (C \cap (M_G \times N_G)), A_{G+C}(i) = j$ et $A_{G+C}(i) = A_G(i)$ sinon. L'utilité de cette allocation est par construction $U_p(A_{G+C}(T_p)) = V_p(A(G)) + w(C)$, ce qui contredit $w(C) > 0$ (puisque V_p considère toujours l'optimal)

- si $w(C) < 0$: nous pouvons effectuer l'opération inverse à partir de $A_{G'}$ pour créer l'allocation $A_{G'-C}$, en posant $\forall (i, j) \in (C \cap (N_G \times M_G)), A_{G'+C}(i) = j$ et $A_{G'+C}(j) = A_G(i)$ sinon. L'utilité de cette allocation est par construction $U_p(A_{G'-C}(T_p)) = V_p(A(G')) - w(C)$ ce qui contredit $w(C) < 0$.

Puisque tout cycle $C \in S$ est de poids nul, cela veut dire que le chemin d'échanges $S_{j_1, j_2} \in S$ appliqué seul sur l'allocation A_G conduit également à une allocation optimale des objets de $A(G')$ aux agents de T_p . Une telle allocation est donc toujours atteignable par un chemin d'échanges, et comme elle est optimale, elle le sera par les chemins d'échanges canoniques.

Donc appliquer un chemin d'échanges canoniques nous donne une allocation optimale. □

Maintenant que nous avons établi ces résultats, nous pouvons détailler la méthode utilisée pour calculer les k meilleurs sous-groupes de T_q de taille $|T_p|$. Nous considérons le sous-groupe G^* qui maximise $V_p(A(G))$ parmi les sous-groupe de T_q de taille $|T_p|$. Nous construisons le graphe Gr_{G^*} correspondant, puis nous listons les couples (j_1, j_2) tels que $j_1 \in T_q \setminus G^*$ et $j_2 \in G^*$. Nous calculons leurs chemins d'échanges canoniques (via un algorithme de plus court chemin), puis nous choisissons le couple (j_1^0, j_2^0) dont le chemin canonique est de poids maximum. Le sous-groupe G_0 est alors de poids maximum parmi les sous-groupes atteignables à partir de G par un seul échange. Il nous reste à montrer que G_0 a le second plus grand poids parmi tous les sous-groupes.

Pour cela, nous allons montrer que le sous-groupe ayant le second plus grand poids est accessible à partir du sous-groupe de plus grand poids par un seul échange.

Lemme 3.4.1

Pour une instance I contenant les types T_p et T_q tels que $|T_p| < |T_q|$, soit le sous-groupe G^* qui maximise $V_p(A(G))$ parmi les $G \subset T_q$ tel que $|G| = |T_p|$. Alors il existe $j_1 \in T_q \setminus G^*$ et $j_2 \in G^*$ tel que $G_0 = (G^* \setminus \{j_2\}) \cup \{j_1\}$ soit le sous-groupe de deuxième plus grande utilité $V_p(A(G))$ parmi les $G \subset T_q$ tels que $|G| = |T_p|$.

Proof. Supposons que G_0 (le sous-groupe de deuxième plus grande utilité $V_p(A(G))$ parmi les $G \subset T_q$ tel que $|G| = |T_p|$) soit à distance $l > 1$ de G^* , c'est-à-dire qu'il existe $j_{1,1}, \dots, j_{1,l}$ dans $T_q \setminus G^*$ et $j_{2,1}, \dots, j_{2,l}$ dans G^* tel que $G_0 = (G^* \setminus \{j_{2,1}, \dots, j_{2,l}\}) \cup \{j_{1,1}, \dots, j_{1,l}\}$.

Nous considérons le graphe d'allocation $Gr_{G^*} = (N_{G^*}, M_{G^*}, E_{G^*})$ et nous définissons l'ensemble d'arêtes $S_{G^* \rightarrow G_0} \subset E_{G^*}$ contenant l'ensemble des arêtes $(A_{G_0}(i), i)$ et $(i, A_{G^*}(i))$ pour tout agent $i \in T_p$ tel que $A_{G^*}(i) \neq A_{G_0}(i)$.

La suite se basera sur le même raisonnement que la preuve de la propriété 3.4.6. Comme chaque agent est affecté à un unique objet dans chaque allocation, chaque agent qui n'est pas dans $j_{1,1}, \dots, j_{1,l}$ ou $j_{2,1}, \dots, j_{2,l}$ sera présent dans exactement deux arêtes de

$S_{G^* \rightarrow G_0}$ ou zéro. Par conséquent, $S_{G^* \rightarrow G_0}$ sera constitué de l chemins d'échanges canoniques disjoints $S_{j_{1,1}, j_{2,1}}, \dots, S_{j_{1,l}, j_{2,l}}$ (à permutation près) et de cycles de poids zéro (sinon on peut les appliquer à G^* ou à G_0).

Nous savons également, par construction, que :

$$\begin{aligned} V_p(A(G^*)) &= V_p(A(G_0)) + w(S_{G^* \rightarrow G_0}) \\ V_p(A(G^*)) &= V_p(A(G_0)) + \sum_{e \in S_{G^* \rightarrow G_0}} w_e \\ V_p(A(G^*)) &= V_p(A(G_0)) + \sum_{k=1}^l \sum_{e \in S_{j_{1,k}, j_{2,k}}} w_e \\ V_p(A(G^*)) &= V_p(A(G_0)) + \sum_{k=1}^l w(S_{j_{1,k}, j_{2,k}}) \end{aligned}$$

Puisque les chemins d'échanges sont disjoints, nous pouvons appliquer n'importe lequel à G^* . Or G^* est optimal, donc tous ces chemins sont de poids négatifs ou nuls. Si il existe deux chemins d'échanges de poids non nul S_1 et S_2 , alors il existe un sous-groupe G_1 obtenu en appliquant S_1 à G^* tel que $V_p(A(G_1)) = V_p(A(G^*)) + w(S_1) = V_p(A(G_0)) - w(S_2)$, ce qui est en contradiction avec le fait que G_0 ait la deuxième plus grande utilité parmi les sous-groupes. Par conséquent, il existe au plus un chemin S_1 de poids non nul. On peut alors obtenir, en appliquant ce chemin à G^* , un sous-groupe de même utilité de G_0 mais atteignable en un seul échange à partir de G^* . \square

Par conséquent, la procédure décrite plus haut nous permet de trouver le sous-groupe ayant la deuxième plus grande utilité.

Une fois que nous avons une telle méthode, nous allons l'utiliser pour générer les k -meilleurs sous-groupes, en utilisant l'algorithme décrit ci-dessous.

Nous construisons un arbre tel que chaque sommet contiennent deux ensembles d'agents H et F et deux sous groupes G et G' tel que $H \subseteq G$, $F \cap G = \emptyset$, $H \subseteq G'$ et $F \cap G' = \emptyset$. H représentera les agents obligatoires, F les agents interdits, G le sous-groupe de meilleure utilité contenant H et ne contenant pas F , et G' le sous-groupe de deuxième meilleure utilité contenant H et ne contenant pas F .

La racine est définie avec $H = \emptyset$, $F = \emptyset$, $G = G^*$ le sous-groupe de meilleure utilité et $G' = G_0$ le sous-groupe de deuxième meilleure utilité atteignable à partir de G^* avec un seul échange (tel que décrit précédemment).

Pour développer une feuille (H, F, G, G') , nous considérons l'unique $j \in G \setminus G'$ existant, puis nous transformons la feuille (H, F, G, G') en nœud interne ayant pour fils les feuilles $(H \cup \{j\}, F, G, G_1)$ et $(H, F \cup \{j\}, G', G_2)$ tels que G_1 soit le sous-groupe de deuxième meilleure utilité contenant les agents $H \cup \{j\}$ et ne contenant pas les agents F , et que G_2 soit le sous-groupe de deuxième meilleure utilité contenant les agents H et ne contenant pas les agents $F \cup \{j\}$. Ces sous-groupes sont générés en recherchant le

chemin d'échanges canonique de poids maximum entre un agent de $T_q \setminus (G \cup F)$ (resp. $T_q \setminus (G \cup F \cup \{j\})$) et un agent de $G \setminus H$ (resp. $G \setminus (H \cup \{j\})$) comme expliqué précédemment.

L'algorithme consiste à développer à chaque itération la feuille dont le deuxième meilleur sous-groupe est d'utilité maximum parmi les sous-groupes des feuilles. Par construction, quelle que soit l'étape, les feuilles de l'arbre constitueront une partition de l'ensemble des sous-groupes, et indiqueront les deux meilleurs sous-groupes de chaque partition. À l'itération i , l'arbre contient $i - 1$ noeuds internes dont les sous-groupes G ont les $i - 1$ plus grandes utilités, le G' de plus grande utilité parmi les feuilles est le sous-groupe de rang i pour l'utilité.

Le pseudo-code de cet algorithme est :

```

Création de la racine contenant  $(\emptyset, \emptyset, G^*, G_0)$  où  $G^*$  est le sous-groupe de
meilleur utilité et  $G_0$  est le sous-groupe de deuxième meilleur utilité ;
Poser  $i=2$  ;
Retenir  $G_0$  comme le sous-groupe de  $i$ -ème meilleur utilité ;
tant que  $i < k$  faire
    Sélectionner la feuille  $(H, F, G, G')$  contenant le sous-groupe de  $i$ -ème
meilleur utilité en dernier membre  $G'$  ;
    Sélectionner l'unique  $j$  tel que  $\{j\} = G \setminus G'$  ;
    Transforme la feuille en noeud interne ;
    Lui ajouter la feuille  $(H \cup \{j\}, F, G, G_1)$  où  $G_1$  est le sous-groupe de deuxième
meilleure utilité parmi ceux contenant  $H \cup \{j\}$  et ne contenant pas  $F$  ;
    Lui ajouter la feuille  $(H, F \cup \{j\}, G', G_2)$  où  $G_2$  est le sous-groupe de deuxième
meilleure utilité parmi ceux contenant  $H$  et ne contenant pas  $F \cup \{j\}$  ;
    Poser  $i=i+1$  ;
    Sélectionner le sous-groupe de plus grande utilité parmi les quatrièmes
membres des feuilles, puis le retenir comme le sous-groupe de  $i$ -ème plus
grande utilité;
fin
    
```

Algorithme 1 : Algorithme de calcul des k meilleurs lots

Calculer le meilleur sous-groupe peut être fait en temps polynomial (problème d'allocation optimale), calculer le deuxième meilleur sous-groupe également (problème de plus court chemin). Cependant, ces opérations doivent être répétées k fois, et k lui-même n'est pas nécessairement polynomial en la taille des groupes. Par conséquent, la complexité de cet algorithme est en $O(\text{poly}(n, k))$.

3.5 Retour sur la notion de monotonie

Lorsque nous avons mis en place nos axiomes, nous avons souhaité représenter la monotonie.

La monotonie est souvent utilisée dans des cadres divers car elle permet de cerner le comportement d'un mécanisme autrement difficile à prédire. En particulier, il peut être très utile dans la conception d'algorithme de savoir comment se comportera un mécanisme (ici une définition d'envie) lorsque' on augmente une instance en ajoutant des agents ou des objets.

La difficulté est de trouver une définition de la monotonie qui soit pertinente dans le cadre de définition d'envie que nous avons proposée. Nous avons précédemment utilisé l'axiome 3.3.4, cependant il existe d'autres approches de la notion, et nous souhaiterions explorer les alternatives.

Dans un deuxième temps, nous nous intéresserons à deux définitions de monotonie utilisées dans la littérature : la monotonie de population (introduite par Thomson [1983]) et la monotonie de ressources (décrite par Moulin et Thomson [1988]). Il nous faut cependant noter que ces notions se réfèrent non pas aux définitions d'envie mais aux mécanismes d'allocation, nous chercherons donc si il est possible à un algorithme minimisant l'envie pour au moins un sous-groupe de vérifier ces notions.

3.5.1 Monotonie de l'envie

Les notions de monotonie que nous allons considérer seront basées sur l'idée d'ajouter un agent et un objet à l'un des deux types comparés. Nous allons représenter ces axiomes par des flèches ayant différentes directions, une vers le haut ou le bas (\uparrow ou \downarrow) et une vers la droite ou la gauche (\rightarrow ou \leftarrow). Il y aura deux possibilités sur le type choisi pour recevoir le couple (agent, objet) :

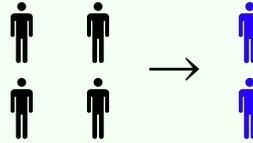
- soit nous ajoutons l'agent et l'objet au type qui se compare T_p . Nous représentons ce cas par une flèche vers la gauche \leftarrow .
- soit nous ajoutons l'agent et l'objet au type qui est comparé T_q . Nous représentons ce cas par une flèche vers la droite \rightarrow .

De même, nous considérerons deux cas possibles pour l'utilité que le nouvel objet apporte aux agents de T_p :

- Soit elle est plus grande que toutes les autres utilités, nous représenterons ce cas par une flèche vers le haut \uparrow .
- Soit elle est plus petite que toutes les autres utilités, nous représenterons ce cas par une flèche vers le bas \downarrow .

Exemple 3.5.1

Nous allons illustrer chaque possibilité.
Nous considérerons un groupe de 4 agents (en noir) se comparant à un groupe de 2 agents (en bleu).

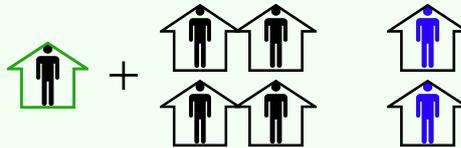


Supposons un profil de préférences de la forme suivante :

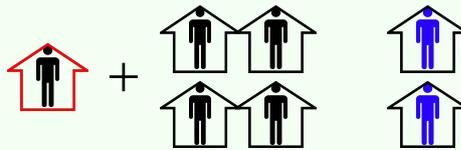


Nous avons alors les cas suivants :

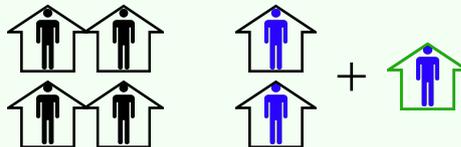
(\leftarrow, \uparrow) : la situation du groupe qui se compare s'améliore



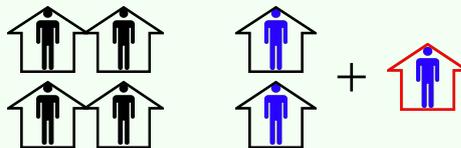
(\leftarrow, \downarrow) : la situation du groupe qui compare empire



(\rightarrow, \uparrow) : la situation du groupe qui est comparé s'améliore



$(\rightarrow, \downarrow)$: la situation du groupe qui est comparé empire



Cela nous conduit à définir 4 axiomes de monotonie différents :

- (\leftarrow, \uparrow) : Si le groupe T_p n'envie pas le groupe T_q , et que l'on ajoute à T_p un agent et un objet associé tel que cet agent obtienne une meilleure utilité de son objet que de n'importe quel objet de T_q , alors ce T_p modifié ne doit toujours pas envier T_q .

- (\leftarrow, \downarrow) : Si le groupe T_p envie le groupe T_q , et que l'on ajoute à T_p un agent et un objet associé tel que cet agent obtienne de son objet une utilité inférieure à celle qu'il obtiendrait de n'importe quel objet de T_q , alors ce T_p modifié doit toujours envier T_q .
- (\rightarrow, \uparrow) : Si le groupe T_p envie le groupe T_q , et que l'on ajoute à T_q un agent et un objet associé tel que tout agent de T_p obtiendrait de cet objet un plus grande utilité que celle de leur objet actuel, alors T_p doit toujours envier ce T_q modifié.
- $(\rightarrow, \downarrow)$: Si le groupe T_p n'envie pas le groupe T_q , et que l'on ajoute à T_q un agent et un objet associé tel que tout agent de T_p obtiendrait de cet objet une plus petite utilité que celle de leur objet actuel, alors T_p ne doit toujours pas envier ce T_q modifié.

Nous pouvons alors nous demander quelles notions d'envie parmi celles que nous avons présentées précédemment vérifient ces axiomes. Les résultats sont présentés dans le tableau suivant :

	(\rightarrow, \uparrow)	$(\rightarrow, \downarrow)$	(\leftarrow, \uparrow)	(\leftarrow, \downarrow)
TEF	Oui	Non	Oui	Non
WEF	Non	Non	Non	Non
A & W	Oui	Oui	Oui	Oui
l'envie pour au moins un sous-groupe	Oui	Non	Non	Oui
l'envie pour tout sous-groupe	Non	Non	Non	Non

Preuves pour TEF. • (\rightarrow, \uparrow) : déjà prouvé.

- $(\rightarrow, \downarrow)$: on considère une instance où $T_p = \{a_1, a_2\}$ et $T_q = \{a_3, a_4\}$, et $\mathcal{M} = \{h_1, h_2, h_3, h_4\}$ tel que : $\forall a, u_a(h_1) = u_a(h_2) = 5$, $u_a(h_3) = 7$ et $u_a(h_4) = 0$. On considère l'allocation A telle que $A_p = \{h_1, h_2\}$ et $A_q = \{h_3, h_4\}$ (on suppose que pour tout i , $A(a_i) = h_i$). On rajoute à T_q l'agent a_0 alloué au bien h_0 tel que $u_{a_0}(h_0) = 4$. L'axiome interdit T_p d'envier ce T_q modifié mais $V_p(\{h_3, h_4, h_5\}) = 7 + 4 = 11 > V_p(A_p) = 10$ donc l'axiome est enfreint.
- (\leftarrow, \uparrow) : Quels que soient T_p et T_q , si l'on pose $T'_p = T_p \cup \{a_0\}$ et $A'_p = A_p \cup \{h_0\}$, alors on sait que $V_{T'_p}(A'_p) - V_p(A_p) \geq u_{a_0}(h_0)$ (on ne sait pas si l'allocation optimale de A'_p à T'_p alloue réellement h_0 à a_0 , mais elle doit être au moins aussi bonne que si elle le faisait). Par hypothèse sur la qualité de h_0 , on a $u_{a_0}(h_0) > V_{T'_p}(A_q) - V_p(A_q)$ (puisque toute allocation des biens de A_q à T'_p va aura au mieux une valeur de $V_p(A_q)$ sur T_p , et une valeur strictement inférieure à $u_{a_0}(h_0)$ sur a_0).
Donc $V_{T'_p}(A'_p) - V_p(A_p) > V_{T'_p}(A_q) - V_p(A_q)$, ce qui nous donne $V_{T'_p}(A'_p) > V_{T'_p}(A_q)$: T'_p ne peut pas envier T_q .

- (\leftarrow, \downarrow) : on considère une instance où $T_p = \{a_1, a_2\}$ et $T_q = \{a_3, a_4\}$, et $\mathcal{M} = \{h_1, h_2, h_3, h_4\}$ tel que : $\forall a \in \{a_1, a_2\}, u_a(h_1) = u_a(h_2) = 5$, $u_a(h_3) = 7$ et $u_a(h_4) = 7$. On considère l'allocation A telle que $A_p = \{h_1, h_2\}$ et $A_q = \{h_3, h_4\}$ (on suppose que pour tout i , $A(a_i) = h_i$). Dans cette situation, T_p envie T_q puisque $V_p(A_p) = 10 < V_p(A_q) = 14$. On pose $T'_p = T_p \cup \{a_0\}$ et $A'_p = A_p \cup \{h_0\}$ avec a_0 et h_0 tel que $u_{a_0}(h_0) = 5$, et $u_{a_0}(h_3) = u_{a_0}(h_4) = 7$ (les autres utilités ne sont pas importantes). Dans cette situation, l'axiome forcerait T'_p à envier T_q , or $V_{T'_p}(A'_p) = 15 > V_{T'_p}(A_q) = 14$: T'_p n'envie pas T_q . □

Preuves pour WEF.

(\rightarrow, \uparrow) : déjà prouvé.

(\rightarrow, \downarrow) : contre-exemple : on considère une instance où $T_p = \{a_0, \dots, a_{10}\}$ et $T_q = \{a_{11}, \dots, a_{20}\}$, $\mathcal{M} = \{h_0, \dots, h_{20}\}$ et des fonctions d'utilité telles que :

- $u_{a_0}(h_0) = 10$
- $\forall j \in \llbracket 11, 20 \rrbracket, u_{a_0}(h_j) = 0$
- $\forall i \in \llbracket 1, 10 \rrbracket, u_i(h_i) = 1$
- $\forall i \in \llbracket 1, 10 \rrbracket, \forall j \in \llbracket 11, 20 \rrbracket, u_i(h_j) = 1, 5$

Les autres utilités ne sont pas importantes et sont supposées nulles. On considère l'allocation A telle que $A_p = \{h_0, \dots, h_{10}\}$ et $A_q = \{h_{11}, \dots, h_{20}\}$. On a alors $\frac{V_p(A_p)}{|T_p|} = \frac{20}{11} > \frac{V_p(A_q)}{|T_q|} = \frac{15}{10}$, donc T_p n'envie pas T_q . Nous ajoutons maintenant à T_q l'agent a_{21} et l'objet h_{21} tel que $u_{a_0}(h_{21}) = 9$ et $\forall i \in \llbracket 1, 10 \rrbracket, u_i(h_{21}) = 0$. L'axiome obligerait alors T_p à ne pas envier $T'_q = T_q \cup \{a_{21}\}$. Or, $\frac{V_p(A_p)}{|T_p|} = \frac{20}{11} < \frac{V_p(A_q \cup \{h_{21}\})}{|T'_q|} = \frac{24}{11}$, donc T_p envie T'_q .

(\leftarrow, \uparrow) : contre-exemple : on considère une instance où $T_p = \{a_1\}$ et $T_q = \{a_2, a_3\}$, $\mathcal{M} = \{h_1, h_2, h_3\}$ tel que $u_{a_1}(h_1) = 5$ et $u_{a_1}(h_2) = u_{a_1}(h_3) = 10$.

On suppose que le système est dans l'allocation A telle que $A_p = \{h_1\}$ and $A_q = \{h_2, h_3\}$. Dans cette allocation, T_p n'envie pas T_q , puisque $\frac{V_p(A_p)}{|T_p|} = \frac{5}{1} = \frac{V_p(A_q)}{|T'_q|} = \frac{10}{2}$.

Nous ajoutons maintenant à T_p l'agent a_0 et l'objet h_0 tel que $u_{a_1}(h_0) = 10$, et que a_0 ait la même fonction d'utilité que a_1 .

L'axiome interdit à $T'_p = T_p \cup \{a_0\}$ d'envier T_q , or $\frac{V'_p(A'_p)}{|T'_p|} = \frac{15}{2} < \frac{V'_p(A_q)}{|T_q|} = \frac{20}{2}$. Par conséquent, l'axiome est enfreint.

(\leftarrow, \downarrow) : contre-exemple : on considère une instance où $T_p = \{a_0\}$ et $T_q = \{a_1, \dots, a_5\}$, $\mathcal{M} = \{h_0, \dots, h_5\}$ et des fonctions d'utilités telles que $\forall j \in \llbracket 1, 5 \rrbracket, u_{a_0}(h_j) = 10$ et $u_{a_0}(h_0) = 1$. On considère l'allocation A telle que $A_p = \{h_0\}$ et $A_q = \{h_1, \dots, h_5\}$. On a alors $\frac{V_p(A_p)}{|T_p|} = \frac{1}{1} < \frac{V_p(A_q)}{|T_q|} = \frac{10}{5} = 2$, donc T_p envie T_q . On pose maintenant $T'_p = T_p \cup \{a_6\}$ et $A'_p = A_p \cup \{h_6\}$ où

a_6 est tel que $u_{a_6}(h_6) = 9$ et $\forall j \in [[1, 5]]$, $u_{a_6}(h_j) = 10$. Dans ce cas, l'axiome obligerait T_p' à être envieux de T_q , cependant $\frac{V_{T_p'}(A_p')}{|T_p'|} = \frac{10}{2} = 5 > \frac{V_{T_p'}(A_q)}{|T_q|} = \frac{20}{5} = 4$, donc T_p' n'envie pas T_q : l'axiome est enfreint. \square

Preuve pour la mesure d'A&W. • (\rightarrow, \uparrow) : déjà prouvé.

- $(\rightarrow, \downarrow)$: Quel que soit T_p et T_q tel que T_p n'envie pas T_q , si l'on pose $T_q' = T_q \cup \{a_0\}$ et $A_q' = A_q \cup \{h_0\}$ tel que $\forall i \in T_p, u_i(A(i)) \geq u_i(h_0)$, alors on a :

$$\sum_{i \in T_p} u_i(A(i)) \geq \sum_{i \in T_p} u_i(h_0)$$

Or puisque T_p n'envie pas T_q nous avons :

$$\frac{1}{|T_p|} \sum_{i \in T_p} u_i(A(i)) \geq \frac{1}{|T_p| \cdot |T_q|} \sum_{i \in T_p} \sum_{j \in T_q} u_i(A(j))$$

$$|T_q| \sum_{i \in T_p} u_i(A(i)) \geq \sum_{i \in T_p} \sum_{j \in T_q} u_i(A(j))$$

d'où, en additionnant avec l'équation pour h_0 :

$$\begin{aligned} \left(|T_q| \sum_{i \in T_p} u_i(A(i)) \right) + \sum_{i \in T_p} u_i(A(i)) &\geq \left(\sum_{i \in T_p} \sum_{j \in T_q} u_i(A(j)) \right) + \sum_{i \in T_p} u_i(A(a_0)) \\ (|T_q| + 1) \sum_{i \in T_p} u_i(A(i)) &\geq \sum_{i \in T_p} \sum_{j \in T_q \cup \{a_0\}} u_i(A(j)) \end{aligned}$$

D'où, puisque $|T_q| + 1 = |T_q'|$:

$$\frac{1}{|T_p|} \sum_{i \in T_p} u_i(A(i)) \geq \frac{1}{|T_p| \cdot |T_q'|} \sum_{i \in T_p} \sum_{j \in T_q'} u_i(A(j))$$

Donc T_p n'envie pas T_q' : l'axiome est respecté.

- (\leftarrow, \uparrow) : quels que soient T_p et T_q tels que T_p n'envie pas T_q , si l'on pose $T_p' = T_p \cup \{a_0\}$ et $A_p' = A_p \cup \{h_0\}$, tel que $\forall j \in T_q, u_{a_0}(h_0) \geq u_{a_0}(A(j))$, alors on a :

$$\sum_{j \in T_q} u_{a_0}(h_0) = |T_q| \cdot u_{a_0}(h_0) \geq \sum_{j \in T_q} u_{a_0}(A(j))$$

Or puisque T_p n'envie pas T_q nous avons :

$$\frac{1}{|T_p|} \sum_{i \in T_p} u_i(A(i)) \geq \frac{1}{|T_p| \cdot |T_q|} \sum_{i \in T_p} \sum_{j \in T_q} u_i(A(j))$$

$$|T_q| \sum_{i \in T_p} u_i(A(i)) \geq \sum_{i \in T_p} \sum_{j \in T_q} u_i(A(j))$$

d'où, en additionnant avec l'équation pour h_0 :

$$\left(|T_q| \sum_{i \in T_p} u_i(A(i)) \right) + |T_q| \cdot u_{a_0}(h_0) \geq \left(\sum_{i \in T_p} \sum_{j \in T_q} u_i(A(j)) \right) + \sum_{j \in T_q} u_{a_0}(A(j))$$

$$|T_q| \sum_{i \in T_p \cup \{a_0\}} u_i(A(i)) \geq \sum_{i \in T_p \cup \{a_0\}} \sum_{j \in T_q} u_i(A(j))$$

d'où:

$$\frac{1}{|T_p'|} \sum_{i \in T_p'} u_i(A(i)) \geq \frac{1}{|T_p'| \cdot |T_q|} \sum_{i \in T_p'} \sum_{j \in T_q} u_i(A(j))$$

Donc T_p' n'envie pas T_q : l'axiome est respecté.

- (\leftarrow, \downarrow) : Quels que soient T_p et T_q tels que T_p envie T_q , si l'on pose $T_p' = T_p \cup \{a_0\}$ et $A_p' = A_p \cup \{h_0\}$, tel que $\forall j \in T_q, u_{a_0}(h_0) < u_{a_0}(A(j))$, alors on a :

$$\sum_{j \in T_q} u_{a_0}(h_0) = |T_q| \cdot u_{a_0}(h_0) < \sum_{j \in T_q} u_{a_0}(A(j))$$

Or puisque T_p envie T_q nous avons :

$$\frac{1}{|T_p|} \sum_{i \in T_p} u_i(A(i)) < \frac{1}{|T_p| \cdot |T_q|} \sum_{i \in T_p} \sum_{j \in T_q} u_i(A(j))$$

$$|T_q| \sum_{i \in T_p} u_i(A(i)) < \sum_{i \in T_p} \sum_{j \in T_q} u_i(A(j))$$

d'où, en additionnant avec l'équation pour h_0 :

$$\left(|T_q| \sum_{i \in T_p} u_i(A(i)) \right) + |T_q| \cdot u_{a_0}(h_0) < \left(\sum_{i \in T_p} \sum_{j \in T_q} u_i(A(j)) \right) + \sum_{j \in T_q} u_{a_0}(A(j))$$

$$|T_q| \sum_{i \in T_p \cup \{a_0\}} u_i(A(i)) < \sum_{i \in T_p \cup \{a_0\}} \sum_{j \in T_q} u_i(A(j))$$

d'où:

$$\frac{1}{|T_p'|} \sum_{i \in T_p'} u_i(A(i)) < \frac{1}{|T_p'| \cdot |T_q|} \sum_{i \in T_p'} \sum_{j \in T_q} u_i(A(j))$$

Donc T_p' envie T_q : l'axiome est respecté. □

Preuves pour l'envie pour au moins un sous-groupe.

(\rightarrow, \uparrow) : déjà prouvé.

$(\rightarrow, \downarrow)$: on considère une instance où $T_p = \{a_1, a_2\}$ et $T_q = \{a_3, a_4\}$, $\mathcal{M} = \{h_1, h_2, h_3, h_4\}$ tel que pour tout agent a , $u_a(h_1) = u_a(h_2) = 5$, $u_a(h_3) = 7$ et $u_a(h_4) = 0$. On considère l'allocation A telle que $A_p = \{h_1, h_2\}$ et $A_q = \{h_3, h_4\}$ (on suppose que pour tout i , $A(a_i) = h_i$). Dans cette allocation, T_p n'envie pas T_q (par l'axiome 3.3.1). On pose maintenant $T'_q = T_q \cup \{a_0\}$ et $A'_q = A_q \cup \{h_0\}$, tel que pour tout agent a , $u_a(h_0) = 4$. L'axiome interdirait à T_p d'envier T'_q , or $V_p(\{h_3, h_4\}) = 11 > V_p(A_p)$, donc T_p envie T'_q pour l'envie pour au moins un sous-groupe : l'axiome est enfreint.

(\leftarrow, \uparrow) : on considère une instance où $T_p = \{a_1, a_2\}$ et $T_q = \{a_3, a_4\}$, $\mathcal{M} = \{h_1, h_2, h_3, h_4\}$ tel que pour tout agent a , $u_a(h_3) = u_a(h_4) = 5$, $u_a(h_1) = 10$ et $u_a(h_2) = 0$. On considère l'allocation A telle que $A_p = \{h_1, h_2\}$ et $A_q = \{h_3, h_4\}$ (on suppose que pour tout i , $A(a_i) = h_i$). Dans cette allocation, T_p n'envie pas T_q (par l'axiome 3.3.1). On pose maintenant $T'_p = T_p \cup \{a_0\}$, $A'_p = A_p \cup \{h_0\}$, $A(a_0) = h_0$, et pour tout agent a , $u_a(h_0) = 6$. L'axiome interdirait à T'_p d'envier T_q , or pour $G = \{a_0, a_2\}$, on a $V_G(A(G)) = 6 < V_G(A_q) = 10$, donc T_p envie T'_q pour l'envie pour au moins un sous-groupe : l'axiome est enfreint.

(\leftarrow, \downarrow) : quels que soient T_p et T_q tels que T_p envie T_q , nous allons ajouter à T_p un agent a_0 alloué à un objet h_0 . Nous posons $T'_p = T_p \cup \{a_0\}$ et $A'_p = A_p \cup \{h_0\}$. Nous allons considérer deux cas différents en fonction de la taille des groupes :

- si $|T_p| < |T_q|$: alors il existe $G \subseteq T_q$ tel que $|G| = |T_p|$ et $V_p(A_p) < V_p(A(G))$. Dans ce cas, nous pouvons considérer $G' = G \cup \{a\}$ pour n'importe quel $a \in T_q \setminus G$. Nous aurons alors, par hypothèse, $V_{T'_p}(A'_p) - V_p(A_p) = u_{a_0}(h_0) < u_{a_0}(a) \leq V_{T'_p}(A(G')) - V_p(A(G))$ d'où $V_{T'_p}(A'_p) < V_{T'_p}(A(G'))$ avec $|G'| = |T'_p|$. Donc T'_p envie T_q .
- si $|T_p| > |T_q|$: alors il existe $G \subseteq T_p$ tel que $|G| = |T_q|$ et $V_G(A(G)) < V_G(A_q)$. Dans ce cas, $G \subseteq T'_p$ et est tel que $|G| = |T_q|$, donc le résultat est immédiat.

□

Preuves pour l'envie pour tout sous-groupe.

(\rightarrow, \uparrow) : déjà prouvé.

$(\rightarrow, \downarrow)$: on considère une instance où $T_p = \{a_1, a_2\}$ et $T_q = \{a_3\}$, $\mathcal{M} = \{h_1, h_2, h_3\}$ et $u_{a_1}(h_1) = u_{a_2}(h_2) = 1$, $u_{a_1}(h_3) = 0$ et $u_{a_2}(h_3) = 1000$. On considère l'allocation A qui à l'agent a_i alloue l'objet h_i ($A_p = \{h_1, h_2\}$ et $A_q = \{h_3\}$). Dans cette allocation, T_p n'envie pas T_q puisque $V_{\{a_1\}}(\{h_3\}) = 0 < V_{\{a_1\}}(\{h_1\}) = 1$. On rajoute à T_q l'agent a_0 et l'objet h_0 tel que $u_{a_1}(h_0) = u_{a_2}(h_0) = 0$. L'axiome demande que T_p n'envie pas $T'_q = T_q \cup \{a_0\}$, or l'axiome 3.3.1 l'oblige puisque $V_p(A_p) = 2 < V_p(A_q \cup \{h_0\}) = 1000$.

(\leftarrow, \uparrow) : on considère une instance où $T_p = \{a_1\}$ et $T_q = \{a_2, a_3\}$, $\mathcal{M} = \{h_1, h_2, h_3\}$ et $u_{a_1}(h_1) = 1$, $u_{a_1}(h_2) = 0$ et $u_{a_1}(h_3) = 1000$. On considère l'allocation A qui à l'agent a_i alloue l'objet h_i ($A_p = \{h_1\}$ et $A_q = \{h_2, h_3\}$). Dans cette allocation, T_p n'envie pas T_q puisque $V_{\{a_1\}}(\{h_2\}) = 0 < V_{\{a_1\}}(\{h_1\}) = 1$. On rajoute à T_p l'agent a_0 et à A_p l'objet h_0 tel que

$u_{a_0}(h_0) = 1$ et $u_{a_0}(h_2) = u_{a_0}(h_3) = 0$. L'axiome demande que T_p n'envie toujours pas T_q , or l'axiome 3.3.1 l'oblige puisque $V_{T_p \cup \{a_0\}}(A_p \cup \{h_0\}) = 2 < V_{T_p \cup \{a_0\}}(A_q) = 1000$.

(\leftarrow, \downarrow) : on considère une instance où $T_p = \{a_1, a_2\}$ et $T_q = \{a_3, a_4\}$, $\mathcal{M} = \{h_1, h_2, h_3, h_4\}$ et les fonctions d'utilités sont telles que pour tout agent $a \in T_p$, $u_a(h_1) = 0$, $u_a(h_2) = 10$, $u_a(h_3) = 6$ et $u_a(h_4) = 6$. On considère l'allocation A qui à chaque agent a_i alloue l'objet h_i ($A_p = \{h_1, h_2\}$ et $A_q = \{h_3, h_4\}$), alors T_p envie T_q (par l'axiome 3.3.1). On rajoute à T_p l'agent a_0 et à A_p l'objet h_0 tel que $A(a_0) = h_0$ et $u_a(h_0) = 4$ pour tout $a \in T_p$. L'axiome demande que $T'_p = T_p \cup \{a_0\}$ n'envie pas T_q , or $V_{\{a_0, a_2\}}(\{h_0, h_2\}) = 14 > V_{\{a_0, a_2\}}(A_q) = 12$, donc la comparaison pour le sous groupe $G = \{h_0, h_2\} \subset T_p$ ne résulte pas en envie : T'_p envie T_q . \square

Les différentes notions d'envie se comportent de manières très différentes face à ces axiomes. Certaines n'en vérifient aucune, et seule la mesure d'A&W parvient à les vérifier toutes. Nous pouvons alors nous demander si il est possible de vérifier l'axiome 3.3.1 en même temps que les axiomes de monotonie.

3.5.2 Monotonie de population et de ressources

Les concepts de monotonie de population (introduit par Thomson [1983]) et de monotonie de ressources (décrit par Moulin et Thomson [1988]) sont une autre manière de considérer la notion de monotonie. Elle s'applique cette fois aux mécanismes de génération d'allocations. La monotonie de population formalise l'idée que si le nombre d'agents augmente alors que la quantité de ressources reste identique, alors le mécanisme ne doit pas allouer aux agents de meilleurs lots de ressources. La monotonie de ressources formalise l'idée que si la quantité de ressources augmente alors que le nombre d'agents reste identique, le mécanisme ne doit pas allouer aux agents de moins bons lots de ressources.

Ces notions peuvent être adaptées à notre cadre de la manière suivante :

Définition 3.5.1: Monotonie de population

Un mécanisme d'allocation est dit **population-monotone** si et seulement si : pour toute instance I contenant les types T_1, \dots, T_k ($k \in \mathbb{N}$), et toute instance I' obtenue à partir de I en ajoutant un type T_{k+1} , si le mécanisme génère l'allocation A quand appliqué à I et l'allocation A' quand appliqué à I' , alors $\forall p \in [[1, k]]$, $V_p(A_p) \geq V_p(A'_p)$.

Définition 3.5.2: Monotonie de ressources

Un mécanisme d'allocation est dit **ressource-monotone** si et seulement si : pour toute instance I contenant les types T_1, \dots, T_k ($k \in \mathbb{N}$) et l'ensemble d'objets $\mathcal{M} = \{h_1, \dots, h_m\}$ ($m \in \mathbb{N}$), et toute instance I' obtenue à partir de I en ajoutant un objet h_{m+1} à l'ensemble \mathcal{M} , si le mécanisme génère l'allocation A quand appliqué à I et l'allocation A' quand appliqué à I' , alors $\forall p \in [[1, k]]$, $V_p(A_p) \leq V_p(A'_p)$.

Malheureusement, nous pouvons montrer que ces notions sont difficiles à réconcilier avec le degré d'envie.

Proposition 3.5.1

Tout mécanisme générant des allocations qui minimisent la somme des degrés d'envie ou le maximum des degrés d'envie ne peut pas être population-monotone.

Preuve. Soit une instance I avec les types $T_1 = \{a_1\}$ et $T_2 = \{a_2, \dots, a_{100}\}$, $\mathcal{M} = \{h_1, \dots, h_{101}\}$ et des fonctions d'utilité telles que :

- $u_{a_1}(h_1) = 10$
- $u_{a_1}(h_2) = 11$
- $\forall a \in T_2, u_a(h_1) = 0$
- $\forall a \in T_2, u_a(h_2) = 0, 2$
- $u_{a_1}(h_3) = u_{a_1}(h_4) = 0.01$
- $\forall a \in T_2, u_a(h_3) = u_a(h_4) = 0.1$
- $\forall i > 4, u_{a_1}(h_i) = 0$
- $\forall i > 4, u_{a_1}(h_i) = 0.3$

L'allocation A qui minimise le degré d'envie (que ce soit le maximum ou la somme) est celle qui affecte à a_i l'objet h_i , pour tout i . Le degré d'envie est alors de $\frac{1}{100}$ de T_1 vers T_2 , et de 0 de T_2 vers T_1 . On crée maintenant l'instance I' à partir de I en rajoutant le type $T_0 = \{a_0\}$ tel que :

- $u_{a_0}(h_1) = 10$
- $u_{a_0}(h_2) = 10$
- $u_{a_0}(h_3) = u_{a_0}(h_4) = 0.1$
- $\forall i > 4, u_{a_0}(h_i) = 0$

La nouvelle allocation A' qui minimise le degré d'envie (que ce soit le maximum ou la somme) est telle que $A'_1 = \{h_2\}$, $A'_0 = \{h_1\}$ et A'_2 contient le reste des objets. Dans cette allocation, T_1 et T_0 sont sans envie, tandis que T_2 a un degré d'envie de 0.02 envers T_1 .

Cependant, $V_{T_1}(A_1) < V_{T_1}(A'_1)$, la situation de T_1 s'est donc améliorée. □

Proposition 3.5.2

Tout mécanisme générant des allocations qui minimisent la somme des degrés d'envie ou le maximum des degrés d'envie ne peut pas être ressource-monotone.

Preuve. Considérons l'instance I' construite dans la preuve précédente. Rappelons que dans l'allocation A' minimisant le degré d'envie (que ce soit le maximum ou la somme), $A'_1 = \{h_2\}$, $A'_0 = \{h_1\}$ et A'_2 contient le reste des objets.

Nous construisons l'instance I^* en ajoutant à \mathcal{M} l'objet h_0 tel que :

- $u_{a_1}(h_0) = 10$
- $u_{a_0}(h_0) = 0$
- $\forall a \in T_2, u_a(h_0) = 0$

La nouvelle allocation I^* qui minimise le degré d'envie (que ce soit le maximum ou la somme) est l'allocation A^* telle que $A^*_1 = \{h_0\}$, $A^*_3 = \{h_2\}$ et A^*_2 contient le reste des objets. Dans cette allocation, les degrés d'envie sont nuls sauf le degré d'envie de T_1 envers T_2 , qui vaut 0.01. Or, $V_{T_1}(A'_1) > V_{T_1}(A^*_1)$, la situation de T_1 a empiré. □

3.6 Conclusion du chapitre

Notre recherche d'une notion d'envie adaptée à notre modèle nous a amené à établir quatre axiomes, chacun représentant une propriété intuitive que nous souhaitions retrouver dans notre notion d'envie. Cependant, il s'est avéré difficile de satisfaire tous ces axiomes en même temps : en particulier, nous avons montré qu'aucune des notions d'envie utilisées dans la littérature ne satisfait les 4 axiomes à la fois. Par conséquent, nous nous sommes attelé à la création d'une nouvelle famille de notion d'envie, basée sur l'idée de comparer le plus petit des deux types à des sous-groupes de l'autre type, les différences entre chaque notion reposant sur le nombre de comparaisons. Nous avons ensuite analysé ces notions en utilisant nos axiomes. Si toutes les nouvelles notions ne satisfont pas tous les axiomes, il en ressort que la notion de l'envie pour au moins un sous-groupe, calculable en temps polynomial, les vérifie tous. Pour finir, nous nous sommes penchés plus en détails sur la notion de monotonie et les différentes significations possibles du terme dans notre contexte. Nous avons, au travers de ces définitions, ré-examiné les notions d'envie évoquées précédemment.

Les perspectives pour de prochains travaux seraient, dans un premier temps, de déterminer avec certitude la complexité de calcul de l'envie pour tout sous-groupe.

Nous n'avons pour l'instant que réussi à réduire le calcul à la résolution d'un problème d'optimisation quadratique, ce qui suggérerait que ce problème soit NP-difficile. Cependant, il nous faudrait trouver une réduction qui prouve cette complexité. Il n'est de plus pas exclu que ce problème soit approximable en temps polynomial. Les axiomes que nous avons choisi de retenir ont été construits pour être simples et intuitifs. Par conséquent, bien qu'ils suffisent pour discriminer les différentes notions d'envie présentées, ils ne forment pas une caractérisation complète de la notion que nous avons construite. Il serait donc envisageable d'essayer de construire un ensemble d'axiomes plus forts et plus précis qui forment une caractérisation complète de l'envie pour au moins un sous-groupe (ou même de la famille de notions d'envie). Enfin, au cours de ce chapitre, nous avons présenté de nombreux axiomes, et même si tous ne sont pas d'importance équivalente, tous représentent une propriété souhaitable pour une notion d'envie. Or, la notion que nous avançons, l'envie pour au moins un sous-groupe, ne les satisfait pas tous, étant incompatible avec plusieurs axiomes de monotonie. Il serait alors pertinent de vérifier si il est possible de concevoir une notion d'envie qui vérifie tous les axiomes (y compris ceux de monotonie), tout en permettant à d'éventuels mécanismes de minimisation de cette envie de respecter la monotonie de ressources et de population.

Les travaux présentés dans ce chapitre, à l'exception de la dernière partie, ont donné lieu à une publication à la conférence ECAI (Gross-Humbert *et al.* [2023]).

Chapitre 4

Minimisation du degré d'envie entre types d'agents

4.1 Introduction

Nous avons introduit au chapitre précédent une définition d'envie adaptée à notre modèle et satisfaisant un certain nombre d'axiomes souhaitables. Malheureusement, nous avons aussi montré qu'une allocation sans-envie n'existe pas toujours, et peut-être difficile à calculer.

Une manière de gérer ce problème est de considérer une relaxation de la définition d'envie. Par exemple, Budish [2011] introduit la notion d'*absence d'envie à un bien près* (EF1), qui autorise les agents à être envieux du moment qu'il reste possible supprimer cette envie en retirant un objet. Une relaxation plus importante a été introduit par Manurangsi et Suksompong [2022], l'*absence d'envie à c bien près* (EF c), qui étend la notion précédente pour autoriser le retrait d'ensemble de jusqu'à c objets.

Une autre approche présente dans la littérature consiste à quantifier l'envie par une valeur que l'on cherchera à minimiser. Il s'agit d'une idée introduite par Feldman et Kirman [1974], qui ont d'abord cherché à compter le nombre de paires d'agents envieux, puis ont décidé de mesurer l'envie dans chacune de ces paires et de la sommer. Ce concept a été repris par Chaudhuri [1986], qui mesurent de manière différente l'envie d'une paire, puis par Diamantaras et Thomson [1990], qui décident, pour quantifier l'envie totale d'une allocation, plutôt que de sommer les mesures d'envie sur toutes les paires, de seulement retenir la plus grande d'entre elles.

Suivant ces concepts, nous souhaiterions déterminer une manière de quantifier notre envie, qu'il soit possible soit de chercher à minimiser soit de comparer à un seuil pour définir une relaxation.

Nous commencerons par définir le degré d'envie, puis nous proposerons deux méthodes d'approximation possible. Enfin, nous discuterons d'une série d'expériences effectuées pour tester l'une de ces méthodes.

4.2 Modèle

Le modèle que nous allons considérer dans ce chapitre sera le même que dans le chapitre précédent. Nous rappelons rapidement les structures et notations introduites dans le chapitre 3 :

- Une instance est constituée d'un ensemble d'agents \mathcal{N} partitionné en types T_1, T_2, \dots, T_k , d'un ensemble d'objets \mathcal{M} et d'une fonction d'utilité u_i pour chaque agent $i \in \mathcal{N}$.
- les objets sont alloués aux types, puis les types allouent leurs lots d'objets à leurs agents de manière à maximiser le bien-être social utilitariste.
- l'utilité d'un lot d'objet pour un type est le bien-être social utilitariste résultant de l'allocation de ce lot à ce type.

Les explications sur le fonctionnement du modèle et les motivations derrière les choix ont déjà été effectués dans le chapitre 3, nous ne reviendrons donc pas dessus.

Le précédent chapitre introduit dans la section 3.4 une notion appelée l'envie pour au moins un sous-groupe définie par :

Définition 4.2.1: l'envie pour au moins un sous-groupe

Pour une instance I dans une allocation A , nous dirons que le type T_p envie le type T_q si et seulement si au moins l'une des conditions suivantes est vérifiée :

- $|T_p| \leq |T_q|$ et $V_p(A_p) < V_p(A(G))$ pour au moins un sous-groupe $G \subseteq T_q$ tel que $|G| = |T_p|$.
- $|T_p| \geq |T_q|$ et $V_G(A(G)) < V_G(A_q)$ pour au moins un sous-groupe $G \subset T_p$ tel que $|G| = |T_q|$.

Comme nous l'avons vu dans la section 3.4, calculer une allocation sans-envie pour cette notion est NP-difficile.

Un autre défaut de l'envie pour au moins un sous-groupe est qu'il est très facile pour une allocation d'en produire, et que dans de nombreux cas, il sera évidemment impossible de trouver une allocation sans-envie.

Plaçons-nous par exemple dans le cas d'utilités binaires communes à tous les agents, si un type d'agent ne contient qu'un seul agent, alors si il n'obtient pas d'objet d'utilité 1 il sera envieux de tous les types ayant une utilité non nulle, et si il en obtient un il sera envié par tous les types ayant une utilité inférieure à leur taille. Même si l'on souhaitait utiliser une relaxation de type "à un bien près", on ne ferait que reporter le problème sur un type à deux agents, qui provoqueraient de l'envie dès que l'on s'éloigne de plus d'un cran des allocation extrême (types d'utilités maximum ou nulle). Par conséquent, il nous faut une relaxation beaucoup plus importante pour être utilisable dans le cas général.

4.3 Degré d'envie

4.3.1 Définition

Nous introduisons le degré d'envie, qui consiste, plutôt que de chercher l'existence d'un sous-groupe de taille égale dont la comparaison résulterait en envie, à compter le nombre de tels sous-groupes et à calculer leur proportion parmi le nombre total de sous-groupes. Formellement :

Définition 4.3.1: Degré d'envie

Soit deux types T_p et T_q et une allocation A , le degré d'envie d_{pq}^A de T_p envers T_q est défini comme :

$$d_{pq}^A = \begin{cases} \frac{|\{G \subseteq T_q : |G|=|T_p| \& V_p(A_p) < V_p(A(G))\}|}{|\{G \subseteq T_q : |G|=|T_p|\}|} & \text{si } |T_p| \leq |T_q| \\ \frac{|\{G \subseteq T_p : |G|=|T_q| \& V_G(A(G)) < V_G(A_q)\}|}{|\{G \subseteq T_p : |G|=|T_q|\}|} & \text{sinon} \end{cases}$$

Exemple 4.3.1: Exemple de calcul de degré d'envie

Considérons une instance I contenant un ensemble d'agents \mathcal{N} partitionnés en deux types $T_1 = \{a_1, a_2\}$ et $T_2 = \{a_3, a_4, a_5\}$, et un ensemble d'objets $\mathcal{M} = \{h_1, h_2, h_3, h_4, h_5\}$. Nous posons les fonctions d'utilité :

	h_1	h_2	h_3	h_4	h_5
a_1	1	1	1	1	0
a_2	0	0	0	1	0
a_3	0	1	1	0	0
a_4	1	0	0	0	0
a_5	1	0	0	0	0

Considérons l'allocation A telle que $A_1 = \{h_1, h_2\}$ et $A_2 = \{h_3, h_4, h_5\}$. Les allocations individuelles seront alors $A(a_1) = h_1$, $A(a_2) = h_2$, $A(a_3) = h_3$, $A(a_4) = h_4$ et $A(a_5) = h_5$.

Dans cet exemple, T_1 va se comparer à chaque sous-groupe G de 2 agents de T_2 :

- pour $G = \{a_3, a_4\}$, $V_1(A_1) = 1 < V_1(A(G)) = 2$: T_1 envie G .
- pour $G = \{a_3, a_5\}$, $V_1(A_1) = 1 = V_1(A(G)) = 1$: pas d'envie.
- pour $G = \{a_4, a_5\}$, $V_1(A_1) = 1 = V_1(A(G)) = 1$: pas d'envie.

Donc le degré d'envie de T_1 envers T_2 dans l'allocation A vaut : $d_{1,2}^A = \frac{1}{3}$

De l'autre côté, chaque sous-groupe G de 2 agents de T_2 va se comparer à T_1 :

- pour $G = \{a_3, a_4\}$, $V_G(A_G) = 1 < V_G(A_1) = 2$: G envie T_1 .
- pour $G = \{a_3, a_5\}$, $V_G(A_G) = 1 < V_G(A_1) = 2$: G envie T_1 .
- pour $G = \{a_4, a_5\}$, $V_G(A_G) = 0 < V_G(A_1) = 1$: G envie T_1 .

Donc le degré d'envie de T_2 envers T_1 dans l'allocation A vaut : $d_{2,1}^A = 1$

Notons que l'envie pour au moins un sous-groupe revient à déterminer si le degré d'envie est différent de zéro, et que l'envie pour tout sous-groupe revient à vérifier si le degré d'envie est égal à 1.

4.3.2 Complexité de calcul

La première considération sur cette mesure est de savoir si il est possible de la calculer en temps raisonnable : il nous faut donc vérifier sa complexité de calcul.

Pour cela, nous définissons d'abord formellement le problème considéré.

Définition 4.3.2: Problème de comptage des sous-groupes enviés

Soit une instance I contenant les types T_p et T_q tels que $|T_p| < |T_q|$ et une allocation A sur cette instance.

Combien existe-t-il de sous-groupe $G \subseteq T_q$ tels que $V_p(A_p) < V_p(A(G))$?

Nous allons montrer que ce problème est $\#P$ -complet, où $\#P$ est la classe de complexité des problèmes de comptage associés aux problèmes NP-complet introduite par Valiant [1979b] (par exemple, le problème SAT consiste à déterminer si une formule logique peut être rendue vraie par une solution, tandis que le problème $\#SAT$ consiste à compter le nombre de solutions qui la rendent vraie).

Proposition 4.3.1: Complexité du problème de comptage des sous-groupes enviés

Le problème de comptage des sous-groupes enviés est $\#P$ -complet, et ce même si tous les agents partagent la même fonction d'utilité.

Proof. Cette preuve est composée de deux parties : montrer que le problème est dans $\#P$, et montrer qu'il est $\#P$ -difficile.

Il est facile de montrer qu'il est dans $\#P$ puisque le problème de décision associé peut être résolu en temps polynomial par un algorithme non-déterministe qui choisit un sous-groupe $G \subseteq T_q$, calcule l'utilité $V_p(A(G))$ et vérifie si cette dernière est suffisamment grande pour que G soit envié.

Pour montrer que ce problème est $\#P$ -difficile, nous allons faire une réduction depuis le problème $\#SubsetSum$, le problème de comptage associé au problème de la

somme des sous-ensembles. Nous savons que ce dernier est $\#P$ -complet car il existe des réductions du problème 3SAT au problème de Partition (Garey et Johnson [1979]) et du problème de Partition au problème Subset Sum (Karp [1972]) qui conservent le nombre de solution. Or, le problème de comptage #3SAT est $\#P$ -complet (Valiant [1979a]). Le problème #SubsetSum peut être défini de la manière suivante : soit un ensemble d'entiers $S = \{s_1, \dots, s_f\}$ ($f \in \mathbb{N}$) et un entier T , combien existe-t-il de sous-ensembles de S dont la somme vaut T ?

Considérons une instance de ce problème, nous allons construire deux instances du problème de comptage des sous-groupes enviés I et I' . Nous définissons l'instance I comme ayant deux types T_1 et T_2 tels que $|T_1| = f$ et $|T_2| = 2f$, et un ensemble d'objet $\mathcal{M} = \{h_1, \dots, h_{2f}\}$. Nous définissons les fonctions d'utilité des agents de T_1 de la manière suivante :

- $\forall a \in T_1, \forall i \leq f, u_a(h_i) = s_f$
- $\forall a \in T_1, \forall i > f, u_a(h_i) = \frac{T}{f}$

Puisque les fonctions d'utilités sont identiques entre les agents, l'utilité ne dépendra pas des agents dans la suite. Nous considérons l'allocation A telle que $A_1 = \{h_{f+1}, \dots, h_{2f}\}$ et $A_2 = \{h_1, \dots, h_f\}$. Nous avons alors $V_1(A_1) = \frac{T}{f} \cdot f = T$. Pour qu'un sous-groupe $G \subseteq T_2$ tel que $|G| = |T_1| = f$ soit envié, il faut donc que $V_1(A(G)) = \sum_{h \in A(G)} u(h) = \sum_{i, h_i \in A(G)} s_i > T$. Par conséquent, à chaque groupe G vérifiant ces contraintes correspond un sous-ensemble $S^+ \subseteq S$ tel que $\sum_{s \in S^+} s > T$. Résoudre le problème de comptage des sous-groupes enviés sur I nous indique donc le nombre de sous-groupes de S dont la somme est strictement supérieure à T .

Nous définissons l'instance I' de manière similaire sauf que les valeurs d'utilité sont définies par :

- $\forall a \in T_1, \forall i \leq f, u_a(h_i) = s_f$
- $\forall a \in T_1, \forall i > f, u_a(h_i) = \frac{T}{f} - \frac{1}{f}$

La différence est que dans I' nous avons $V_1(A_1) = T - 1$, ce qui veut dire que résoudre le problème de comptage des sous-groupes enviés sur I' nous indique le nombre de sous-groupes de S dont la somme est strictement supérieure à $T - 1$ au lieu de T . Or, tous les éléments de S sont des entiers, donc cela revient à calculer le nombre de sous-groupe de S dont la somme est supérieure ou égale à T .

Ayant obtenu ces deux valeurs, il ne nous reste plus qu'à soustraire le nombre de sous-groupes de S dont la somme est strictement supérieure à T au nombre de sous-groupes de S dont la somme est supérieure ou égale à T pour obtenir le nombre de sous-groupes de S dont la somme vaut exactement T . Par conséquent, résoudre le problème de comptage des sous-groupes enviés nous permet de résoudre le problème de comptage de la sommes des sous-ensembles. Puisque cette dernière est $\#P$ -difficile, le problème de comptage des sous-groupes enviés l'est également. \square

De la même manière, nous pouvons définir le problème inverse :

Définition 4.3.3: Problème de comptage des sous-groupes envieux

Soit une instance I contenant les types T_p et T_q tels que $|T_p| > |T_q|$ et une allocation A sur cette instance.

Combien existe-t-il de sous-groupe $G \subseteq T_p$ tels que $V_G(A(G)) < V_G(A_q)$?

Ce problème est symétrique et peut être résolu de la même manière.

Calculer le degré d'envie revient à résoudre un problème de comptage soit des sous-groupes enviés soit des sous-groupes envieux, par conséquent, il s'agit d'un problème $\#P$ -complet. Puisque la valeur exacte du degré d'envie est trop difficile à calculer, nous allons devoir trouver un moyen de calculer une approximation en temps raisonnable.

4.4 Approximation du degré d'envie

Nous allons maintenant proposer deux méthodes pour approximer le degré d'envie dans le cas où les agents d'un même groupe partagent la même fonction d'utilité. Ces deux méthodes présentent chacune leur avantages et leur défauts :

- La première est déterministe et sa complexité théorique est très inférieure, en revanche, elle requiert que l'on se limite à des utilités rationnelles.
- La seconde a une complexité théorique très importante, mais nous verrons lors des expériences que sa complexité pratique est raisonnable. De plus, elle peut être étendue naturellement au cas général (mais sans garantie théorique de fonctionnement).

4.4.1 Réduction à une variante du problème de sac à dos

Cette méthode repose sur l'utilisation d'un algorithme d'approximation du problème du $\#$ Exact M -items Knapsack (problème de comptage associé au problème du "sac-à-dos à exactement M objets"). Un tel algorithme a été proposé par Triommatitis et Pagourtzis [2020], calculant une approximation à un ϵ près en temps $O(\frac{n^4}{\epsilon} \log(\frac{n}{\epsilon}))$. Il suffit donc de transformer une instance du problème de comptage des sous-groupes enviés en une instance du problème du $\#$ Exact M -items Knapsack d'une manière qui conserve le nombre de solutions et la précision de l'approximation.

Commençons par poser le problème du $\#$ Exact M -items Knapsack :

Définition 4.4.1: Problème de comptage du sac à dos à exactement M items (#Exact M -items Knapsack)

Soit un ensemble de poids d'objets $W = \{w_1, \dots, w_n\}$ (pour $n \in \mathbb{N}$, et $\forall i \in \llbracket 1, n \rrbracket$, $w_i \in \mathbb{R}_+^*$), une capacité $C \in \mathbb{R}$ et un entier $M \in \llbracket 1, n \rrbracket$.

Combien existe-t-il de sous-groupes $S \subseteq \llbracket 1, n \rrbracket$ tels que :

- $|S| = M$
- $\sum_{i \in S} w_i \leq C$

Nous allons maintenant voir comment réduire des instances du problème des sous-groupes enviés à une instance de ce problème. Soit une instance I dans une allocation A , soit T_p et T_q deux types de cette instance tels que $|T_p| < |T_q|$, soit u_p la fonction d'utilité des agents de T_p . Nous imposons que ces utilités soient rationnelles, et nous supposons sans restriction de généralité qu'elles sont entières (puisque nous pouvons toujours les multiplier par le PPCM de leur dénominateurs pour nous ramener au cas entier, l'algorithme de Triommatis et Pagourtzis [2020] ne dépendant pas des valeurs des poids). Nous numérotions les agents de T_q de la manière suivante : $T_q = \{a_1, a_2, \dots, a_n\}$ où $n = |T_q|$.

Nous posons $w^* = \max_{a \in T_q} u_p(A(a))$, nous définissons l'ensemble des poids W en créant un objet w_i pour chaque $i \in \llbracket 1, n \rrbracket$ en posant $w_i = w^* - u_p(A(a_i))$. Nous posons ensuite $M = |T_p|$ et $C = M * w^* - V_p(A(T_p)) - 1$.

Il existe une bijection évidente entre les sous-groupe $G \subseteq T_q$ tels que $|G| = |T_p|$ et les sous-ensemble $S \subseteq \llbracket 1, n \rrbracket$ tels que $|S| = M$, puisqu'à partir d'un sous-ensemble S on peut prendre l'ensemble $\{a_i \in T_q | i \in S\}$ et vice-versa.

Si $S \subseteq \llbracket 1, n \rrbracket$ est un sous-ensemble tel que $|S| = M$ et $\sum_{i \in S} w_i \leq C$, alors

$$\sum_{i \in S} w^* - u_p(A(a_i)) \leq M * w^* - V_p(A(T_p)) - 1 < M * w^* - V_p(A(T_p))$$

donc, comme $|S| = M$, nous avons

$$w^* M - \sum_{i \in S} u_p(A(a_i)) < M * w^* - V_p(A(T_p))$$

$$\sum_{i \in S} u_p(A(a_i)) > V_p(A(T_p))$$

Donc le sous-groupe G associé à ce sous-ensemble S est envié.

Réciproquement, si nous avons un sous-groupe G tel que

$$\sum_{a_i \in G} u_p(A(a_i)) > V_p(A(T_p))$$

Alors comme toutes les utilités sont entières nous avons :

$$\sum_{a_i \in G} u_p(A(a_i)) \geq V_p(A(T_p)) + 1$$

d'où

$$w^*M - \sum_{a_i \in G} u_p(A(a_i)) \leq M * w^* - V_p(A(T_p)) - 1$$

Donc le sous-ensemble S associé au sous-groupe G vérifie :

$$w^*M - \sum_{i \in S} u_p(A(a_i)) \leq M * w^* - V_p(A(T_p)) - 1$$

Par conséquent, nous pouvons transformer notre instance du problème des sous-groupes enviés en instance de #Exact M -items Knapsack en conservant le même nombre de solutions, le résultat de l'algorithme de Triommatis et Pagourtzis [2020] approximant la solution de ce problème (et sa précision) sera donc également valide pour notre problème.

4.4.2 Approximation par un FPRAS

La deuxième méthode d'approximation utilise un algorithme d'approximation randomisé de type FPRAS (pour Fully Polynomial Randomised Approximation Scheme), c'est-à-dire un algorithme ayant une probabilité d'au moins $\frac{3}{4}$ de renvoyer une valeur dans un ϵ -voisinage de la solution exacte, et ce en temps polynomial en la taille de l'entrée et en $\frac{1}{\epsilon}$.

En nous inspirant des concepts discutés par Jerrum et Sinclair [1996], nous allons concevoir un FPRAS approximant la taille de notre ensemble de solutions Ω contenant deux parties :

1. une méthode d'échantillonnage capable de générer aléatoirement des solutions de Ω en suivant une distribution de probabilités sur Ω connue.
2. une famille d'ensembles $(\Omega_i)_{i \in \llbracket 0, n \rrbracket}$ tels que $\Omega_n \subset \Omega_{n-1} \subset \dots \subset \Omega_0 = \Omega$ et $|\Omega_n| = 1$

L'objectif de cette méthode est d'estimer les ratios successifs $\frac{|\Omega_i|}{|\Omega_{i+1}|}$ à l'aide de la méthode d'échantillonnage : nous générons un ensemble d'échantillon de Ω_0 et vérifions combien sont dans Ω_1 , puis nous générons un ensemble d'échantillon de Ω_1 et vérifions combien sont dans Ω_2 , et ainsi de suite. Une fois tous les ratios estimés, nous pouvons les utiliser pour estimer $|\Omega|^{-1}$ puisque :

$$|\Omega|^{-1} = \frac{|\Omega_0|}{|\Omega_n|} = \frac{|\Omega_0|}{|\Omega_1|} \times \dots \times \frac{|\Omega_{n-1}|}{|\Omega_n|}$$

Bien sûr, puisque nous allons multiplier des estimations entre elles, nous devons faire attention à l'évolution de l'espérance et de la variance. Cependant, les calculs relatifs à ces deux valeurs ont déjà été faits par Jerrum et Sinclair [1996].

La principale difficulté va être de trouver une méthode d'échantillonnage efficace.

4.4.3 Méthode d'échantillonnage

Nous souhaitons construire une méthode qui nous permette d'échantillonner notre espace des solutions (c'est-à-dire l'ensemble des sous-groupes enviés) en temps polynomial en suivant une distribution de probabilités uniforme.

Pour cela, nous allons utiliser une marche aléatoire sur une chaîne de Markov $M(\Omega, P)$ dont l'ensemble d'états sera notre ensemble de solutions Ω . Pour que cette chaîne soit appropriée, elle doit vérifier 3 propriétés :

- Cette chaîne de Markov doit être **irréductible**, ce qui veut dire que nos états doivent être reliés par une relation de voisinage qui garantie que l'ensemble d'états est connexe.
- Cette chaîne de Markov doit avoir une distribution de probabilités stationnaire π (nous souhaitons également que cette distribution soit uniforme).
- Cette chaîne doit être **rapidly mixing**, c'est-à-dire qu'elle doit s'approcher rapidement de sa distribution stationnaire (en particulier, en temps très petit devant $|\Omega|$).

Considérons une instance I contenant les types T_p et T_q (tels que $|T_p| < |T_q|$) dans l'allocation A . Soit Ω l'ensemble des sous-groupes $G \in T_q$ tels que $|G| = T_p$ et $V_p(A_p) < V_p(A(G))$.

Nous définissons notre chaîne de Markov $M(\Omega, P)$ utilisant Ω comme espace d'états en décrivant un processus qui sera utilisé à chaque étape pour changer d'état : en considérant que nous sommes dans l'état $x \in \Omega$,

- avec probabilité $\frac{1}{2}$ nous restons dans l'état x .
- sinon, nous tirons au hasard un agent $a_1 \in x$ et un agent $a_2 \in (T_q \setminus x)$ et considérons le sous-groupe $x' = (x \setminus \{a_1\}) \cup \{a_2\}$: si $x' \in \Omega$ nous passons dans l'état x' , sinon nous restons dans l'état x .

Notre matrice de transition P sera donc :

- $\forall x \in \Omega, \forall x' \text{ voisin de } x, P(x, x') = \frac{1}{2|T_p|(|T_q| - |T_p|)}$
- $\forall x \in \Omega, P(x, x) = 1 - \sum_{x' \text{ voisin de } x} P(x, x')$

où x' est un voisin de x si et seulement si $|x \cap x'| = |T_p| - 1$.

Irréductibilité de la chaîne de Markov

Soit deux sous-groupes enviés g_1 et g_2 de Ω . D'après la notion de voisinage que nous venons d'introduire, g_1 et g_2 sont voisins (et auront $P(g_1, g_2) > 0$) si et seulement si on peut obtenir g_2 à partir de g_1 en retirant un agent de g_1 et en ajoutant un agent de $T_q \setminus g_1$ (et vice-versa).

Pour montrer que notre chaîne de Markov est irréductible, il nous suffit de montrer que pour tout couple de sous-groupes enviés x et y de Ω , il est possible d'atteindre y en partant de x par ajouts et retraits successifs d'agents un par un, en faisant en sorte que tous les sous-groupes d'agents intermédiaires soient aussi enviés (c'est-à-dire dans Ω).

Theorem 4.4.1: Irréductibilité de la chaîne de Markov

Pour la relation de voisinage décrite ci-dessus, la chaîne de Markov sur Ω est irréductible : pour tout $x \in \Omega$ et $y \in \Omega$, il existe une suite d'éléments de Ω x_0, x_1, \dots, x_m pour un $m \in \mathbb{N}$ telle que $x = x_0$, $y = x_m$, et pour tout $i \in \llbracket 0, m-1 \rrbracket$ on ait x_i voisin de x_{i+1} (et réciproquement)

Ce résultat se base sur une construction utilisée par Chegireddy et Hamacher [1987], qui a déjà inspiré l'algorithme de calcul de l'envie pour k sous-groupes dans le chapitre précédent. Par conséquent, nous allons réintroduire ici des structures très semblables, mais nous les utiliserons de manière légèrement différente. Nous mentionnerons à titre indicatif quand l'objet que nous considérerons aura un équivalent dans la description de la méthode de calcul de l'envie pour k sous-groupes. À partir du sous-groupe x , nous définissons le **graphe d'allocation** de x comme dans la définition 3.4.4.

Définition 4.4.2: Graphe d'allocation

Pour une instance I contenant les type T_p et T_q tels que $|T_p| < |T_q|$ et d'un sous-groupe $x \subset T_q$ tel que $|x| = |T_p|$, nous définissons le **graphe d'allocation** G_x comme un graphe bipartite orienté pondéré $G_x = (N_x \cup M_x, E_x)$ avec :

- N_x contient un sommet pour chaque agent de T_p
- M_x contient un sommet pour chaque agent de T_q
- $\forall i \in N_x, \forall j \in M_x, (i, j) \in E_x$ si et seulement si l'allocation du lot d'objets $A(x)$ aux agents de T_p maximisant la somme des utilités (c'est-à-dire celle utilisée pour calculer $V_p(A(x))$) attribue l'objet $A(j)$ à l'agent i . Son poids $w(i, j)$ est alors défini par $w(i, j) = -u_i(A(j))$
- $\forall i \in N_x, \forall j \in M_x, (j, i) \in E_x$ si et seulement si $(i, j) \notin E_x$, et son poids est alors défini par $w(j, i) = u_i(A(j))$.

Nous rappelons que ce graphe représente l'allocation des éléments de x aux agents de T_p , le poids de l'arête $(j, i) \in M_x \times N_x$ représente l'utilité que T_p gagnerait à affecter l'objet $A(j)$ à i (en supposant qu'il soit libre), tandis que le poids de l'arête $(i, j) \in N_x \times M_x$ représente l'utilité que T_p perdrait si l'agent i perdait l'objet $A(j)$.

Par conséquent, nous pouvons représenter les différences entre l'allocation optimale des objets de $A(x)$ aux agents de T_p (que nous appellerons A_x dans la suite) et l'allocation optimale des biens de $A(y)$ à T_p (que nous appellerons A_y dans la suite) par un ensemble d'arêtes $S_{x,y}$ représentant les différences entre les deux allocations, défini par :

- $\forall i \in T_p$ tel que $A_x(i) \neq A_y(i)$, $(i, A_x(i)) \in S_{x,y}$
- $\forall i \in T_p$ tel que $A_y(i) \neq A_x(i)$, $(A_y(i), i) \in S_{x,y}$

Lemme 4.4.1

$S_{x,y}$ est tel que $V_p(A(y)) = V_p(A(x)) + \sum_{e \in S_{x,y}} w_e$

Preuve. Ce lemme est immédiat en posant : $V_p(x) = \sum_{i \in T_p} u_i(A_x(i))$ et $V_p(y) = \sum_{i \in T_p} u_i(A_y(i))$

Donc

$$V_p(y) - V_p(x) = \sum_{i \in \{i | A_x(i) \neq A_y(i)\}} (u_i(A_y(i)) - u_i(A_x(i)))$$

Or $\forall i \in T_p$ tel que $A_x(i) \neq A_y(i)$, nous avons $w_{(i, A_x(i))} = -u_i(A_x(i))$ et $w_{(A_y(i), i)} = u_i(A_y(i))$

Donc

$$\sum_{e \in S_{x,y}} w_e = \sum_{i \in \{i | A_x(i) \neq A_y(i)\}} w_{(i, A_x(i))} + w_{(A_y(i), i)} = \sum_{i \in \{i | A_x(i) \neq A_y(i)\}} u_i(A_y(i)) - u_i(A_x(i))$$

□

Notons également que l'ensemble $S_{x,y}$ vérifie la propriété suivante :

Proposition 4.4.1

L'ensemble $S_{x,y}$ peut-être vu comme un ensemble de chemins sur G_x disjoints de taille paire qui appartiennent à l'une des catégories suivantes :

- des chemins qui vont d'un objet de $y \setminus x$ vers un objet de $x \setminus y$
- des cycles de poids zéro (c'est-à-dire la somme des poids de leur arêtes est de zéro).

Proof. Le fait que les chemins soient disjoints vient du fait que l'allocation A_y alloue chaque objet à un unique agent de T_p , donc par construction si $(j, i) \in S_{x,y}$ alors on ne peut pas avoir $(j, i') \in S_{x,y}$ avec $i \neq i'$. La parité de la taille est due au fait que tous les agents de T_p impliqués perdent un objet pour en recevoir un autre, nous aurons donc pour chaque arête de la forme $(i, j) \in N_x \times M_x$, une arête de la forme $(j, i) \in M_x \times N_x$ et vice-versa. Enfin, par construction, les seules arêtes sans prédécesseur possible seront les arêtes $(j, i) \in M_x \times N_x$ telles que $j \in y \setminus x$, et les seules arêtes sans successeur possible seront les arêtes $(i, j) \in N_x \times M_x$ telles que $j \in x \setminus y$. Il nous reste à montrer qu'il ne peut pas exister de chemin circulaire, cela vient de l'optimalité de A_x et A_y : par l'absurde, si il existe un tel chemin, alors il décrit un échange d'objets de $A(x \cap y)$, par conséquent un tel échange peut être appliqué à A_x pour former A'_x , et l'échange inverse peut être

appliqué à A_y pour former $A'y$. Nous aurons alors $V_p(A'_x) = V_p(A_x) + \sum_{e \in S'} w_e$ et $V_p(A_y) = V_p(A'_y) + \sum_{e \in S'} w_e$, ce qui est possible seulement si $\sum_{e \in S'} w_e = 0$ \square

Notons que nous pouvons voir chaque chemin de $S_{x,y}$ comme un chemin d'échanges, tel que défini dans la définition 3.4.5.

Nous allons maintenant considérer l'ensemble des chemins de $S_{x,y}$ ordonnés par poids décroissants p_1, \dots, p_m (où le poids $w(p)$ du chemin p est la somme des poids des arêtes qui le composent). Nous supposons ici par soucis de simplicité qu'il n'y a pas de cycle de poids zéro, la présence de ces derniers ne changeant que peu la preuve (puisque'ils n'influent ni la composition des sous-groupes ni l'utilité, et peuvent donc être appliqués n'importe quand). Cet ensemble va nous permettre de considérer les sous-groupes $x = x_0, \dots, x_m = y$ où x_k est le sous-groupe obtenu en appliquant les permutations correspondant aux chemins p_1 à p_k et $m = |x| - |x \cap y|$. Notons que chaque couple (x_k, x_{k+1}) est voisin pour la définition donnée plus haut. De plus, en appliquant les permutations correspondant aux chemins à l'allocation A_x , nous obtenons une suite d'allocations $A_x = A_{x_0}, \dots, A_{x_m} = A_y$ où A_{x_k} est l'allocation obtenue en appliquant les permutations correspondant aux chemins p_1 à p_k (ces allocations sont semblables aux allocations définies par les chemins de la définition 3.4.6). Notons que ces allocations ne sont pas forcément optimales pour leur sous-groupe, c'est-à-dire que A_{x_k} n'est pas forcément l'allocation optimale du lot d'objets $A(x_k)$ aux agents de T_p , cependant nous savons que les utilités des allocations optimales des lots d'objet correspondant sont au moins aussi élevées que les leurs : nous aurons toujours $V_p(A(x_k)) \geq \sum_{a \in T_p} u_a(A_{x_k}(a))$ que que soit k .

Or, de manière similaire au lemme 4.4.1, nous pouvons poser :

Lemme 4.4.2

Pour toute allocation A_{x_k} générée en appliquant les permutations correspondant aux chemins p_1 à p_k à A_x , nous avons :

$$\sum_{a \in T_p} u_a(A_{x_k}(a)) = V_p(A(x)) + \sum_{r=0}^k w(p_r)$$

ou encore :

$$\sum_{a \in T_p} u_a(A_{x_k}(a)) = V_p(A(x)) + \sum_{r=0}^k \sum_{e \in p_r} w_e$$

Preuve. La preuve de ce lemme est similaire à celle du lemme 4.4.1 \square

Un corollaire de ce lemme est que l'utilité de l'allocation A_{x_k} vérifie l'égalité :

$$\sum_{a \in T_p} u_a(A_{x_k}(a)) = V_p(A(y)) - \sum_{r=k}^m w(p_r)$$

Or, puisque les poids p_k sont triés par ordre décroissant, nous savons que pour tout $i \in [[1, m]]$ nous avons soit $\forall r \in [[1, k]], w(p_r) \geq 0$ soit $\forall r \in [[i, m]], w(p_r) \leq 0$ (selon si $w(p_k)$ est positif ou négatif).

Par conséquent, nous avons soit $\sum_{r \in [[1, k]]} w(p_r) \geq 0$ soit $\sum_{r \in [[k, m]]} w(p_r) \leq 0$.

Donc nous aurons soit $\sum_{a \in T_p} u_a(A_{x_k}(a)) \geq V_p(A(x))$ soit $\sum_{a \in T_p} u_a(A_{x_k}(a)) \geq V_p(A(y))$

Donc $V_p(A(x_k)) \geq \sum_{a \in T_p} u_a(A_{x_k}(a)) \geq \min(V_p(A(x)), V_p(A(y)))$.

Or x et y sont tous deux des sous-groupes enviés, donc $\min(V_p(A(x)), V_p(A(y))) > V_p(A_p)$, ce qui implique $V_p(x_k) > V_p(A_p)$

Donc x_k est aussi un sous-groupe envié et appartient à Ω .

Donc si x et y sont dans Ω , alors tous les x_i sont dans Ω .

Donc pour tout sous-groupe x et y de Ω , il existe une suite de sous-groupes voisins deux à deux reliant x à y dans Ω : nous avons prouvé le théorème 4.4.1.

Probabilité stationnaire

Comme notre chaîne de Markov est irréductible et est telle que $P(x, x) \geq \frac{1}{2}$ pour tout $x \in \Omega$, nous savons qu'elle admet une probabilité stationnaire π telle que :

$$\forall x, y \in \Omega, \pi(x)P(x, y) = \pi(y)P(y, x)$$

Et donc en particulier pour x et y voisins puisque $P(x, y) = P(y, x)$ nous aurons :

$$\pi(x)P(x, y) = \pi(y)P(y, x) = \pi(x) \frac{1}{2|T_p|(|T_q| - |T_p|)} = \pi(y) \frac{1}{2|T_p|(|T_q| - |T_p|)}$$

Or la chaîne de Markov est irréductible : tous les états sont reliés par des "chemins" de voisins.

Donc $\forall x, y \in \Omega, \pi(x) = \pi(y)$

Donc $\forall x \in \Omega, \pi(x) = \frac{1}{|\Omega|}$

La chaîne de Markov est rapidly mixing

Nous devons maintenant montrer que la chaîne de Markov $M(\Omega, P)$ se rapproche suffisamment vite de la probabilité stationnaire π . Commençons par poser formellement ce que nous allons attendre de cette chaîne de Markov.

Définition 4.4.3: Temps de mélange

Le **temps de mélange** $\tau_X(\varepsilon)$ d'une chaîne de Markov $M(\Omega, P)$ est défini par :

$$\tau_X(\varepsilon) = \min\{t : \forall t' \geq t, \Delta_X(t') < \varepsilon\}$$

où la distance de variation $\Delta_X(t)$ au départ de X au temps t est définie par :

$$\Delta_X(t) = \frac{1}{2} \sum_{Y \in \Omega} |P^t(X, Y) - \pi(Y)|$$

Définition 4.4.4: Rapidly Mixing

Une chaîne de Markov est dite **rapidly mixing** si son temps de mélange $\tau_X(\varepsilon)$ est en $O(\text{poly}(\log(|\Omega|), \log(\varepsilon^{-1})))$

Le résultat que nous allons prouver dans cette partie est :

Theorème 4.4.2

La chaîne de Markov $M(\Omega, P)$ définie au début de la section 4.4 est rapidly mixing.

Preuve. Cette preuve est basée sur une preuve apportée par Guruswami [2016], qui prouve un résultat similaire pour un problème de sac à dos. Cette preuve repose sur le fait que si nous parvenons à définir un flux f sur Ω tel que pour tout $X, Y \in \Omega$ tels que $X \neq Y$ nous avons $\sum_{p \in \mathbb{P}_{X,Y}} f(p) = 1$ où $\mathbb{P}_{X,Y}$ est l'ensemble des chemins sans boucle de X à Y , alors nous aurons :

$$\tau_X(\varepsilon) \leq 2n \frac{C(f)}{|\Omega|} L(f)(n + \ln \varepsilon^{-1})$$

où $C(f)$ est le flux maximum sur les arêtes, et $L(f)$ la longueur du plus long chemin de flux. Il nous suffirait donc de construire un flux avec $C(f) = |\Omega| \text{poly}(|T_q|)$ and $L(f) = \text{poly}(|T_q|)$

Pour cela, nous allons commencer par introduire des définitions utilisée par Guruswami [2016] dont nous aurons besoin dans la suite :

Définition 4.4.5: Permutation l-équilibrée

Soit $\{w_i\}_{i=1}^m$ un ensemble de poids de \mathbb{R} et soit $l \in \mathbb{N}$.

Soit $M = \max_{i \leq m} |w_i|$ et $W = \sum_i w_i$.

Une permutation $\sigma \in S_m$ est dite **l-équilibrée** si et seulement si :

$$\forall k \in \llbracket 1, m \rrbracket, \min(W, 0) - lM \leq \sum_{i=1}^k w_{\sigma(i)} \leq \max(W, 0) + lM$$

Définition 4.4.6: Permutation α -uniforme

Soit σ une variable aléatoire sur S_m et soit $\alpha \in \mathbb{R}$.

σ est une **permutation α -uniforme** si et seulement si :

$$Pr_{\sigma}[\sigma(\{1, 2, \dots, k\}) = U] \leq \alpha \times \binom{m}{k}^{-1}$$

Nous allons également avoir besoin d'un théorème de Guruswami [2016] :

Theorème 4.4.3

Il existe une constante universelle C telle que pour tout $m \in \mathbb{N}$ et tout ensemble de poids réels $\{w_i\}_{i=1}^m$ il existe une permutation sur $\{w_i\}_{i=1}^m$ qui est 7-équilibrée et Cm^2 -uniforme.

De plus, si $|\sum_i w_i| > 15 \max_i |w_i|$, alors il existe une permutation sur $\{w_i\}_{i=1}^m$ qui est 0-équilibrée et Cm^2 -uniforme.

Nous avons encore besoin de prouver un résultat avant de passer à la définition du flux.

Considérons X et Y deux états de Ω . Soit $E_{X \rightarrow Y} \subset ((X \setminus Y) \times (Y \setminus X))$ un ensemble d'échanges d'agents qui appliqué à X , le transforme en Y . Nous nous restreindrons dans la suite aux ensembles $E_{X \rightarrow Y}$ de tailles minimales, c'est-à-dire tels que $|E_{X \rightarrow Y}| = |X| - |X \cap Y|$. Nous savons d'après la preuve de l'irréductibilité de la chaîne de Markov qu'il existe au moins un tel ensemble, et puisque nous nous sommes placé dans le cas de fonctions d'utilités communes entre agents, lequel nous choisissons n'est pas important.

Nous divisons $E_{X \rightarrow Y}$ en deux ensembles H et S tels que H contiennent les 29 échanges de plus grands poids (en valeur absolue) et S les autres échanges (la valeur 29 sera nécessaire pour une équation plus loin dans la preuve). Nous posons $m = |S|$ et $\{w_i\}_{i=1}^m$ les poids des échanges de S , c'est-à-dire les variations d'utilité engendrées par l'échange (par exemple, pour deux agents i et j de T_q , $w_{i,j} = u(A(j)) - u(A(i))$). Nous divisons l'ensemble H en deux ensembles H^- et H^+ qui contiennent respectivement les échanges

de poids strictement négatifs et les échanges de poids positifs ou nuls de H . Pour simplifier les notations, nous étendons la fonction d'utilité u aux ensembles d'échanges, l'utilité d'un ensemble d'échanges est égale à la somme des poids des échanges qui le composent

Nous pouvons maintenant adapter un théorème de Guruswami [2016] à notre cadre :

Theorème 4.4.4

Pour la constante C du théorème 4.4.3, il existe une famille de permutations Cm^2 -uniforme telle que chaque permutation σ de cette famille vérifie, $\forall k \in \llbracket 1, m \rrbracket$,

$$\min\{u(A(Y)) - u(A(X)), 0\} - u(H^+) \leq \sum_{i=1}^k w_{\sigma(i)} \leq \max\{u(A(Y)) - u(A(X)), 0\} - u(H^-)$$

Proof. Nous supposons que $m > 0$ (sinon le résultat est immédiat). Nous rappelons que nous avons $u(A(Y)) = u(A(X)) + u(E_{X \rightarrow Y})$ d'après le lemme 4.4.1 car $E_{X \rightarrow Y}$ peut être vu comme un ensemble de chemins sur le graphe d'allocation G_x (chaque chemin correspondant à un échange). Soit $M = \max_{s \in S} |w_i|$ et $W = \sum_{i \in S} w_i$. En décomposant $E_{X \rightarrow Y}$, nous obtenons :

$$u(A(Y)) = u(A(X)) + u(H^+) + u(H^-) + \sum_{i=1}^m w_i$$

Nous avons donc

$$W = u(A(Y)) - u(A(X)) - u(H^+) - u(H^-)$$

et

$$u(A(Y)) - u(A(X)) = W + u(H^+) + u(H^-)$$

L'encadrement du théorème 4.4.4 que nous cherchons à prouver est donc équivalente à

$$\min\{W + u(H^+) + u(H^-), 0\} - u(H^+) \leq \sum_{i=1}^k w_{\sigma(i)} \leq \max\{W + u(H^+) + u(H^-), 0\} - u(H^-)$$

Pour prouver ce résultat, nous allons devoir considérer deux cas distincts, en fonction de si W est supérieur à $-u(H^+) - u(H^-)$ ou non

- **cas 1** : $W \geq -u(H^+) - u(H^-)$: Dans ce cas, le résultat est équivalent à :

$$-u(H^+) \leq \sum_{i=1}^k w_{\sigma(i)} \leq W + u(H^+)$$

Le théorème 4.4.3 nous garantit l'existence d'une permutation 7-équilibrée, et dans certains cas 0-équilibré, pour l'utiliser dans cette preuve nous allons montrer que nous pouvons toujours choisir un $l \in \{0, 7\}$ tel que $lM \leq \min(u(H^+), W + u(H^+))$. Nous procédons par disjonction des cas sur la valeur de W :

- $|W| > 15M$: dans ce cas nous pouvons prendre $l = 0$, puisque le théorème 4.4.3 nous garantit une permutation 0-équilibrée. Nous aurons évidemment $lM \leq u(H^+)$, si $W \geq 0$ nous aurons $lM \leq W + u(H^+)$ et si $W < 0$ alors comme par hypothèse nous avons $W + u(H^-) \geq -u(H^+)$ alors avec $u(H^-) \leq 0$ et $u(H^+) \geq 0$ nous aurons $W + u(H^+) \geq 0 = lM$.
- $0 \leq W \leq 15M$: dans ce cas nous aurons

$$15M \geq W \geq -u(H^+) - u(H^-)$$

Or par construction $29M \leq u(H^+) - u(H^-)$

En soustrayant les deux équations précédentes, nous obtenons

$$2u(H^+) \geq 14M$$

Nous avons donc déjà $u(H^+) \geq lM$ pour $l = 7$.

Or par hypothèse $W \geq 0$, nous obtenons donc $u(H^+) + W \geq lM$ pour $l = 7$.

- $-15M \leq W < 0$: En soustrayant $W \geq -u(H^+) - u(H^-)$ à $29M \leq u(H^+) - u(H^-)$ nous obtenons $2u(H^+) \geq 29M - W$ et donc :

$$u(H^+) \geq \frac{29M - W}{2} \geq \frac{14M - 2W}{2} = 7M - W$$

Nous avons donc $u(H^+) \geq u(H^+) + W \geq lM$ pour $l = 7$.

Il existe donc toujours une permutation σ qui soit l -équilibrée avec l tel que $lM \leq \min(u(H^+), W + u(H^+))$ Nous avons donc :

$$\forall k \in \llbracket 1, m \rrbracket, \min(W, 0) - lM \leq \sum_{i=1}^k w_{\sigma(i)} \leq \max(W, 0) + lM$$

avec $\min(W, 0) - lM \geq \min(W, 0) - \min(u(H^+), W + u(H^+)) = -u(H^+)$ (et ce, que $W < 0$ ou non) et $\max(W, 0) + lM \leq \max(W, 0) + \min(u(H^+), W + u(H^+)) = W + u(H^+)$ (idem, que $W < 0$ ou non). D'où

$$-u(H^+) \leq \sum_{i=1}^k w_{\sigma(i)} \leq W + u(H^+)$$

- **cas 2** : $W < -u(H^+) - u(H^-)$: Dans ce cas le résultat est équivalent à :

$$W + u(H^-) \leq \sum_{i=1}^k w_{\sigma(i)} \leq -u(H^-)$$

La méthode est très similaire au cas précédent : nous allons montrer que nous pouvons utiliser le théorème 4.4.3 pour obtenir une permutation l -équilibrée où l est tel que $lM \leq \min(-u(H^-), -W - u(H^-))$. Nous procédons par disjonction des cas :

- $|W| > 15M$: dans ce cas le théorème nous autorise $l = 0$. L'inégalité est immédiate si $W < 0$ (puisque $u(H^-) \leq 0$), si $W > 0$ alors comme par hypothèse nous avons $W + u(H^+) < -u(H^-)$ avec $u(H^+) \geq 0$ nous aurons $-W - u(H^-) > u(H^+) \geq 0 = lM$. Donc nous avons bien $lM \leq \min(-u(H^-), -W - u(H^-))$ pour $l = 0$.
- $0 \geq W \geq -15M$: dans ce cas, $-15M \leq W \leq -u(H^+) - u(H^-)$ et comme $29M \leq u(H^+) - u(H^-)$ nous obtenons $-2u(H^-) \geq 14M$. Nous en déduisons donc que $-W - u(H^-) \geq -u(H^-) \geq lM$ pour $l = 7$.
- $0 \leq W \leq 15M$: dans ce cas, nous pouvons soustraire $W < -u(H^+) - u(H^-)$ à $-29M \geq -u(H^+) + u(H^-)$ pour obtenir :

$$-u(H^-) \geq \frac{29M + W}{2} \geq \frac{14M + 2W}{2} = 7M + W$$

D'où $-u(H^-) \geq -u(H^-) - W \geq lM$ pour $l = 7$

Il existe donc toujours une permutation σ qui soit l -équilibrée avec l tel que $lM \leq \min(-u(H^-), -W - u(H^-))$.

Nous avons donc :

$$\forall k \in [[1, m]], \min(W, 0) - lM \leq \sum_{i=1}^k w_{\sigma(i)} \leq \max(W, 0) + lM$$

avec $\min(W, 0) - lM \geq \min(W, 0) - \min(-u(H^-), -W - u(H^-)) = W - u(H^-)$ (que $W < 0$ ou non) et $\max(W, 0) + lm \leq \max(W, 0) + \min(-u(H^-), -W - u(H^-)) = -u(H^-)$ (idem, que $W < 0$ ou non).

D'où :

$$W + u(H^-) \leq \sum_{i=1}^k w_{\sigma(i)} \leq -u(H^-)$$

Nous avons donc prouvé le théorème 4.4.4 dans tous les cas. □

Nous pouvons maintenant définir le flux pour chaque paire d'états de Ω . Soit deux états X et Y de Ω , nous divisons le flux entre chaque famille d'échanges possible. Si $t = |X \cap Y|$ alors il existe $(|T_p| - t)!$ familles (chaque association possible des agents de $X \setminus Y$ aux agents de $Y \setminus X$), le flux sera donc de $\frac{1}{|T_p| - t}$ pour chaque famille. Nous pouvons ensuite, pour chaque famille d'échanges $E_{X \rightarrow Y}$ définissant des ensembles H et S (et donc une famille de poids $\{w_i\}_{i=1}^m$), diviser le flux sur des chemins de flux correspondant à des permutations Cm^2 -uniforme que nous définirons de la manière suivante :

Définition 4.4.7: Chemin de flux

Pour une famille d'échanges $E_{X \rightarrow Y}$ séparée en H^+ , H^- et S , pour $\{w_i\}_{i=1}^m$ les poids des échanges de S et pour σ une permutation Cm^2 -uniforme de la famille garantie par le théorème 4.4.4, nous définissons le chemin de flux en suivant l'algorithme suivant :

- Nous commençons à $k = 0$.
- Tant que $k < m$ faire :
 - si $w_{\sigma(k+1)} < 0$: appliquer l'échange $w_{\sigma(k+1)}$ et incrémenter k si cela ne nous fait pas sortir de Ω , sinon appliquer un échange de H^+ ou annuler un échange de H^- .
 - si $w_{\sigma(k+1)} > 0$: appliquer un échange de H^- ou annuler un échange de H^+ si cela ne nous fait pas sortir de Ω , sinon appliquer l'échanger $w_{\sigma(k+1)}$ et incrémenter k .
- Quand $k = m$, si tous les échanges de H n'ont pas été appliqués : appliquer successivement les échanges de H , en appliquant en priorité ceux de H^- mais en restant dans Ω

Nous utilisons ici l'ensemble d'échanges H pour rester aussi proche que possible de la frontière de Ω . Il nous reste à montrer que nous ne serons jamais forcés de la franchir.

Theorème 4.4.5

Pour toute famille d'échange $E_{X \rightarrow Y}$ et toute permutation σ tel que décrit dans la définition 4.4.7, il est toujours possible de créer un chemin sur Ω en suivant l'algorithme.

Preuve. Pour montrer que le chemin généré est dans Ω il nous suffit de montrer que l'utilité de chaque étape intermédiaire est supérieure à $V_p(A_p)$. Or, $V_p(A_p) \leq \min(V_p(A(X)), V_p(A(Y)))$ puisque X et Y sont dans Ω . De plus, le théorème 4.4.4 nous garantit :

$$\min\{V_p(A(Y)) - V_p(A(X)), 0\} - u(H^+) \leq \sum_{i=1}^k w_{\sigma(i)}$$

$$\min\{V_p(A(Y)), V_p(A(X))\} - u(H^+) \leq \sum_{i=1}^k w_{\sigma(i)} + V_p(A(X))$$

$$V_p(A_p) \leq \min\{V_p(A(Y)), V_p(A(X))\} \leq \sum_{i=1}^k w_{\sigma(i)} + V_p(A(X)) + u(H^+)$$

Donc quelle que soit l'étape k de notre exécution, nous serons obligatoirement dans Ω si tous les échanges de H^+ ont été appliqués et aucun de H^- ne l'est. De plus, comme $\forall h \in H, k \in [[1, m-1]]$, $w_{\sigma(k)} \leq w_h$, nous n'aurons besoin à chaque étape que d'appliquer ou annuler un seul échange de H .

En appliquant le même raisonnement à la borne supérieure, nous pouvons aussi prouver que

$$\sum_{i=1}^k w_{\sigma(i)} + V_p(A(X)) + u(H^-) \leq \max(V_p(A(X)), V_p(A(Y)))$$

□

Nous pouvons maintenant calculer le flux passant par un état quelconque $Z \in \Omega$. Pour cela, nous allons compter le nombre de triplets (X, Y, E) (où X et Y sont des états de Ω et E une famille d'échanges menant de X à Y) pour lesquels un chemin de flux passe par Z .

Pour compter ces triplets, nous allons les associer un à un à des 5-uplets (Z', E', T, U, H') où $Z' \in \Omega$, et E', T, U et H' sont des ensembles d'échanges tels que $|T| \leq 2$, $|H'| \leq 29$ et $T \cup H' \cup U \subseteq E'$. Nous allons établir une injection de l'ensemble des triplets possibles (pour lesquels un chemin de flux passe par Z) vers l'ensemble des 6-uplets possibles, puis sommer le flux sur l'espace d'arrivée.

Soit un triplet (X, Y, E) et un état $Z \in \Omega$ tel que $Z \subseteq X \cup Y$ et que pour tout échange $(i, j) \in E$, $i \in Z$ ou $j \in Z$ (sinon les chemins de flux correspondants à la famille E ne passeront pas par Z). Nous rappelons que $H \subseteq E$ contient les 29 échanges de plus grand poids en valeur absolue de E , et que S contient les autres. Nous allons commencer par nous intéresser à $T \subset H$.

Nous appellerons dans la suite, pour n'importe quel ensemble d'échanges T tel que $\forall (i, j) \in T, i \notin Z \text{ XOR } j \notin Z$, Z_T le sous-groupe (qui n'est pas forcément dans Ω) atteint en partant de Z et en appliquant, pour tout échange $(i, j) \in T$, l'échange (i, j) si $i \in Z$ et (j, i) sinon.

Nous allons montrer le lemme suivant :

Lemme 4.4.3

Pour tout triplet (X, Y, E) dont un chemin de flux passe par Z , il existe $T \subset H$ tel que $|T| \leq 2$ et :

$$V_p(A(Z_T)) \leq \max(V_p(A(X)), V_p(A(Y)))$$

Preuve. Cette preuve se base sur la manière dont nous avons construit les chemins de flux. Nous allons montrer que pour un chemin de flux donné, à chaque étape k les états pour lesquels nous passons sont à une "distance" au plus 2 de la borne souhaitée. Nous savons déjà que $\forall k \in \llbracket 0, m \rrbracket$, $\sum_{i=1}^k w_{\sigma(i)} + V_p(A(X)) + u(H^-) \leq \max(V_p(A(X)), V_p(A(Y)))$, nous savons donc que cette propriété est automatiquement vérifiée dans tous les cas où le nombre d'échanges de H disponible pour faire diminuer l'utilité (c'est-à-dire les échanges de H^- non appliqués et les échanges de H^+ déjà appliqués et donc annulables) est inférieure ou égale à 2. Nous ne regarderons dans la suite que les cas où il reste encore au moins 3 tels échanges de H . Pour un chemin de flux quelconque associé à une permutation σ , nous procédons par récurrence sur l'étape k :

- pour $k = 0$:

Nous partons de X et appliquons d'abord des échanges de H^- jusqu'à ce que cela ne soit plus possible, les états que nous rencontrerons durant cette partie auront nécessairement une utilité inférieure ou égale à $V_p(A(X))$. Nous allons ensuite appliquer $\sigma(1)$, or $\forall h \in H, |w_{\sigma(1)}| \leq |w_h|$ donc n'importe quel échange de H restant suffit à ramener l'utilité en-dessous de $V_p(A(X)) \leq \max(V_p(A(X)), V_p(A(Y)))$.

- pour $0 < k < m$, $w_{\sigma(k+1)} < 0$, en supposant la propriété vrai pour $k - 1$:

Nous allons soit appliquer $\sigma(k+1)$ immédiatement (auquel cas la propriété est immédiate puisque $w_{\sigma(k+1)} < 0$), soit faire remonter l'utilité en appliquant un échange de H^+ ou en annulant un échange de H^- avant d'appliquer $\sigma(k+1)$ (un seul échange $h \in H$ sera requis puisque $|w_h| > w_{\sigma(k+1)}$). Dans ce deuxième cas, faire l'opération inverse (annuler échange de H^+ , ré-appliquer échange de H^-) nous permet immédiatement de vérifier la propriété pour les deux états traversés.

- $0 < k < m$, $w_{\sigma(k+1)} > 0$, en supposant la propriété vrai pour $k - 1$:

Nous allons soit faire descendre l'utilité avec un $h \in H$ (ce qui respecte évidemment la propriété), soit appliquer l'échange $\sigma(k+1)$. Dans ce deuxième cas, nous savons que nous ne pouvons plus utiliser de $h \in H$ pour faire descendre l'utilité sans descendre en-dessous de $V_p(A_p)$, et comme $|w_h| > w_{\sigma(k+1)}$ pour tout $h \in H$, utiliser deux échanges de h permet toujours de faire descendre l'utilité en-dessous de $V_p(A_p) \leq \max(V_p(A(X)), V_p(A(Y)))$, la propriété est donc vérifiée.

- $k = m$:

Lors de cette étape, nous appliquons autant d'échanges de H^- que possible, appliquant des échanges de H^+ sinon. Il nous faut seulement considérer les cas où nous appliquons un échange de H^+ (appliquer un échange de H^- respecte la propriété).

Par construction, si il n'est nécessaire que d'en appliquer un avant de recommencer à appliquer des échanges de H^- , alors annuler cet échange et appliquer un échange de H^- nous fait passer en dessous de $V_p(A_p) \leq \max(V_p(A(X)), V_p(A(Y)))$. Si il est nécessaire d'en ajouter plusieurs avant de pouvoir appliquer un échange de H^- , alors appliquer ce dernier et annuler le dernier échange de H^+ nous fait passer en-dessous de $V_p(A_p) \leq \max(V_p(A(X)), V_p(A(Y)))$. Enfin, si il ne reste plus d'échange de H^- à appliquer, alors appliquer les échanges restant augmentera l'utilité jusqu'à $V_p(A(Y))$, ce qui veut dire que l'utilité sur le chemin sera inférieure ou égale à $V_p(A(Y)) \leq \max(V_p(A(X)), V_p(A(Y)))$.

Donc où que Z soit sur le chemin, on peut descendre en-dessous de $\max(V_p(A(X)), V_p(A(Y)))$ avec au plus deux échanges de H . \square

Nous pouvons donc définir $T \in H$ comme un ensemble d'au plus 2 échanges tels que $V_p(Z_T) \leq \max(V_p(A(X)), V_p(A(Y)))$, notre lemme nous garantit son existence.

Nous pouvons maintenant poser $E' = (\{(i, j) | (i, j) \in E\} \cup \{(j, i) | (i, j) \in E\}) \cap (Z_T \times T_q)$ l'ensemble des échanges entre agents de E réorientés pour qu'ils soient tous applicables à Z_T .

Nous allons construire Z' , que nous définissons par : $Z' = ((X \cup Y) \setminus Z_T) \cup (X \cap Y)$. Essentiellement, Z' contient tous les agents de $X \cup Y$ qui ne sont pas déjà dans Z_T , et les agents qui n'interviennent pas dans les échanges. Cela nous assure que $Z_T \cup Z' = X \cup Y$ et $X \cap Y = Z_T \cap Z'$. La raison pour laquelle nous avons besoin de définir Z_T , est de nous assurer que Z' soit dans Ω , ce qu'il nous faut montrer.

Lemme 4.4.4

Z' tel que défini plus haut est dans Ω , c'est-à-dire que $V_p(A(Z')) > V_p(A_p)$

Preuve. Soit l'ensemble d'échanges $R \subseteq E$ tel qu'appliquer R à X permette d'atteindre Z_T . Remarquons que d'après la définition de Z' , appliquer l'ensemble d'échanges $E \setminus R$ nous permet d'atteindre Z' , puisque Z' contient :

- $X \cap Y$, c'est-à-dire les agents de X et Y qui ne sont pas concernés par les échanges
- $X \cap Z_T$, c'est-à-dire les agents de X qui ont été échangé par R et ne sont plus dans Z_T .
- $Y \cap Z_T$, c'est-à-dire les agents de Y qui doivent encore être échangé par E (et donc n'ont pas été rentré par R)

De plus, appliquer les échanges de R à Z' nous permet d'atteindre Y , de même qu'appliquer les échanges de $E \setminus R$ à Z_T nous permet d'atteindre Y .

Nous avons donc :

$$V_p(A(Z')) = V_p(A(X)) + \sum_{e \in (E \setminus R)} w_e = V_p(A(Y)) - \sum_{e \in R} w_e$$

Mais nous savons que :

$$V_p(A(Z_T)) = V_p(A(X)) + \sum_{e \in (R)} w_e = V_p(A(Y)) - \sum_{e \in E \setminus R} w_e$$

et que :

$$V_p(A(Z_T)) \leq \max(V_p(A(X)), V_p(A(Y)))$$

en réutilisant le lemme 4.4.1, nous avons :

$$V_p(A(Z')) = V_p(A(X)) + \sum_{e \in E} w_e - \sum_{e \in R} w_e + V_p(A(X)) - V_p(A(X))$$

$$V_p(A(Z')) = V_p(A(Y)) + V_p(A(X)) - (V_p(A(X)) + \sum_{e \in R} w_e)$$

$$V_p(A(Z')) = V_p(A(Y)) + V_p(A(X)) - V_p(A(Z_T))$$

$$V_p(A(Z')) \geq V_p(A(Y)) + V_p(A(X)) - \max(V_p(A(X)), V_p(A(Y)))$$

$$V_p(A(Z')) \geq \min(V_p(A(X)), V_p(A(Y))) > V_p(A_p)$$

(puisque X et Y sont tout deux dans Ω) □

Notons également qu'appliquer E' à Z_T nous amène à Z' , par construction : appliquer les échanges qui ont été retournés par rapport à E nous ramène à X , puis appliquer les échanges de $E \cap E'$ nous amène à Z' .

Enfin, il nous reste à définir U et H' , que nous définissons par $U \subseteq S$, $H' \subseteq H$ et $U \cup H' = E \setminus E'$. Concrètement, ces sous-ensembles nous indique quels échanges de E nous permettraient d'atteindre Z_T à partir de X .

Maintenant que nous avons nos 5-uplets, nous pouvons majorer le flux passant par un sommet $Z \in \Omega$. Pour cela nous devons dénombrer les 5-uplets possibles.

Pour créer T , nous n'avons pas d'autre choix que de choisir un ensemble de un ou deux échanges parmi tous les échanges possibles de $Z \times (T_q \setminus Z)$, ce qui nous fait $(|T_p| \cdot (|T_q| - |T_p|))(1 + ((|T_p| - 1) \cdot (|T_q| - |T_p| - 1)))$ possibilités. Nous pouvons ensuite tirer un état $Z' \in \Omega$, ce qui nous fait $|\Omega|$ possibilités. T nous donne Z_T , qui avec Z' nous permet de choisir E' parmi les associations possibles des agents de $Z_T \setminus Z'$ aux agents de $Z' \setminus Z_T$, ce qui nous fait $|T_p|!$ possibilités (en fait $(|T_p| - |Z' \cap Z_T|)!$, mais nous n'avons aucune hypothèse sur $|Z' \cap Z_T|$). Enfin, Il ne nous reste plus qu'à choisir H' et U , le premier ayant 2^{29} possibilités, et le second 2^m que nous préférons décomposer en $\binom{m}{|U|}$ pour chaque valeur de $|U|$.

Nous utilisons ce décompte des 5-uplets pour majorer le nombre de chemins, et nous savons, grâce à la Cm^2 -uniformité de la famille de permutations que nous avons utilisé pour définir nos chemins de flux, et à la répartition du flux sur les $|T_q|!$ familles

d'échanges possibles, que le flux sur chaque chemin de flux est d'au plus $\frac{Cm^2}{\binom{m}{|U|}|T_p|!}$. Nous devons donc additionner les chemins pour chaque valeur de $|U|$

Récapitulons les termes que nous devons multiplier, pour une valeur de $|U|$ donnée :

- $(|T_p|. (|T_q| - |T_p|))(1 + ((|T_p| - 1). (|T_q| - |T_p| - 1)))$ possibilités pour T
- $|\Omega|$ possibilités pour Z'
- $|T_p|!$ pour E' .
- 2^{29} pour H' .
- $\binom{m}{|U|}$ pour U .
- $\frac{Cm^2}{\binom{m}{|U|}|T_p|!}$ pour le flux.

Les variables m et $|T_p|$ sont majorés par $|T_q|$, le flux $C(f)$ est donc bien majoré par une fonction de la forme $|\Omega|poly(|T_q|)$. De plus, grâce à la manière dont nous avons construit les chemins de flux, nous avons $L(f) = 29(|T_q| + 1)$.

La chaîne de Markov $M(\Omega, P)$ est donc bien rapidly mixing. □

4.5 Expériences

Même si la complexité théorique de la méthode d'approximation du degré d'envie qui se base sur un FPRAS est polynomiale, les facteurs constants impliqués sont tellement importants que si cette borne était atteinte, cette méthode ne serait pas exécutable en temps raisonnable, comme démontré par l'étude expérimentale menée par Newman et Vardi [2020] sur un FPRAS similaire. L'objectif de cette partie est donc d'étudier la convergence pratique de la chaîne de Markov, en la testant sur des instances de petites tailles. Une autre limite de notre méthode est qu'elle ne fonctionne que dans le cas de fonctions d'utilité communes entre les agents d'un même groupe, nous essaierons donc de l'appliquer tel quel au cas de fonctions d'utilités individuelles, afin de déterminer si la méthode fonctionne également en pratique dans le cas général.

Les expériences ont été réalisées en générant aléatoirement des instances du problème de calcul du nombre de sous-groupes enviés suivant un paramètre n_p variant entre 4 et 8. Ce paramètre correspond à la taille de T_p , tandis que la taille de T_q est posée à $2n_p$. Les fonctions d'utilité sont tirées aléatoirement : l'utilité que chaque objet confère est tirée selon une distribution de probabilités uniforme entre 0 et 1 (dans le cas des fonctions d'utilité individuelles, l'utilité que chaque objet confère à chaque agent a été tirée de cette manière). À partir de ces fonctions d'utilité, nous avons généré l'ensemble des sous-groupes enviés Ω , ainsi que la chaîne de Markov $M(\Omega, P)$ suivant les spécifications détaillées dans la section 4.4. Nous avons ensuite étudié l'évolution des probabilités de

présence $\pi(t) = \mu_0 P^t$ en partant d'une distribution de probabilités initiale μ_0 égale à 1 pour un état choisi et 0 pour les autres. Nous avons ensuite mesuré la vitesse de convergence vers la distribution stationnaire en retenant les instants t où la distance de variation descend en-dessous de ε pour $\varepsilon = 0.1$ (en rouge) et $\varepsilon = 0.001$ (en bleu).

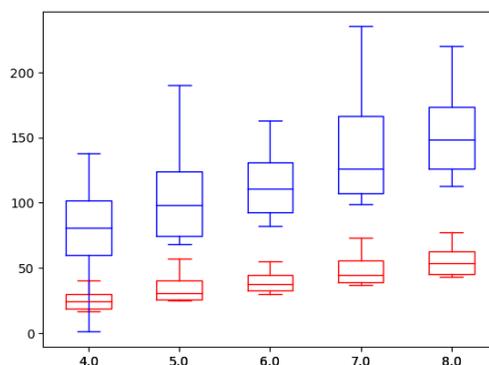


Figure 4.1: Nombre d'étapes avant convergence en fonction de n_p (fonctions d'utilité communes entre agents d'un même type)

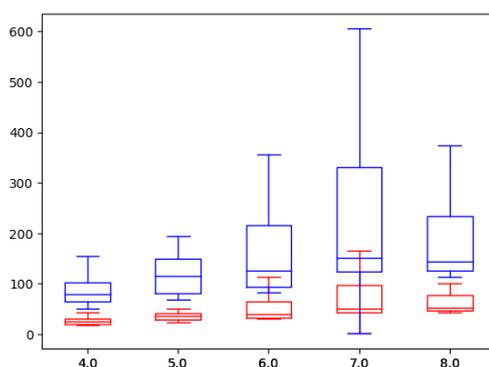


Figure 4.2: Nombre d'étapes avant convergence en fonction de n_p (fonctions d'utilité individuelles)

Nous observons que le nombre d'étapes nécessaires en pratique reste raisonnable sur de petits exemples, en particulier il reste très inférieur à la valeur donnée par la borne théorique, même pour de très petits ε . Pour les deux modèles d'utilité, la convergence est atteinte avec un nombre de pas moyen de moins de 200, et ce nombre semble augmenter linéairement avec la taille de l'instance (à titre de comparaison, Newman et Vardi [2020] ont calculé que le FPRAS qu'ils considéraient nécessitait plusieurs mois de temps CPU, même sur de très petites instances). Notons que dans le cas des fonctions d'utilité individuelles, pour lequel il n'existe pas de borne théorique (ou même de garantie de convergence), non seulement l'algorithme converge toujours mais il reste très inférieur à la borne théorique du cas restreint, même si son temps de convergence

est légèrement supérieur et semble plus sensible aux cas extrêmes. La principale limite pour nos expériences est la taille mémoire, puisque étudier la convergence de la probabilité de présence de la chaîne de Markov requiert de modéliser son espace d'états en entier ainsi que les listes de voisins, cependant le FPRAS lui-même ne requiert ni l'un ni l'autre. Cette expérience suggère donc que cette méthode fonctionne pour des fonctions d'utilités individuelles et des groupes de tailles importantes, même si nous n'avons pas de borne théorique intéressante.

4.6 Conclusion du chapitre

Nous avons introduit dans ce chapitre le degré d'envie, qui permet de mesurer par une valeur numérique l'envie d'un type d'agent pour un autre. Une telle mesure peut par la suite être utilisée soit dans une relaxation, où nous déciderons d'un seuil et imposerons que les valeurs de degré d'envie restent en-dessous, soit dans un problème de minimisation de l'envie, où nous essayerions de garder la somme ou le maximum de ces valeurs sur toute l'instance aussi bas que possible. Le principal obstacle à l'utilisation de cette mesure est la difficulté de calcul, et nous avons montré qu'il est très difficile de calculer la valeur exacte du degré d'envie. Nous avons proposé donc deux méthodes d'estimation dans le cas où les utilités des agents sont identiques au sein d'un type : la première, par réduction au problème du *#Exactly-M-item-knapsack*, relativement efficace, la deuxième, reposant sur un FPRAS, de complexité pratique importante. Nos expériences suggèrent cependant que cette deuxième méthode est bien plus rapide en pratique que la borne théorique ne le laisse supposer. De plus, malgré l'absence de garantie théorique, elle semble fonctionner dans le cas général.

Une première piste d'amélioration serait de retravailler la borne théorique sur le FPRAS. Cette borne préconise un nombre d'étapes très grand, même sur de petits exemples, ce qui ne se retranscrit pas dans les expériences pratiques. Il nous faudrait trouver, pour améliorer cette borne, une nouvelle manière de calculer la convergence d'une chaîne de Markov *rapidly mixing* qui n'utilise pas les résultats sur lesquels reposent la méthode actuelle. Il s'agirait cependant d'une tâche difficile, cette borne étant construite sur les résultats d'autres travaux, l'améliorer demanderait certainement de repartir de zéro. Une autre piste d'amélioration immédiate serait d'étendre cette borne au cas général, où les agents peuvent avoir des fonctions d'utilités différentes.

La continuation naturelle de ces travaux est de chercher des méthodes de minimisation du degré d'envie. Toute utilisation de cette mesure aura pour objectif de la garder aussi basse que possible, il sera donc toujours utile de savoir comment la réduire. De plus, les possibilités de minimisation peuvent nous fournir des indications sur la meilleure manière d'utiliser cette mesure : si il est facile de réduire cette mesure en-dessous d'un certain seuil, l'utiliser dans une relaxation est envisageable, sinon, elle devra être utilisée dans le cadre d'une minimisation sous contrainte.

Une autre inconnue à étudier dans la continuité de la minimisation est la compatibilité de cette notion avec d'autres notions classiques d'efficacité et d'équité telles que l'optimalité de Pareto ou le maximin share. Une question évidente est de savoir si

une allocation minimisant le degré d'envie est nécessairement Pareto-optimale ou respectueuse de la propriété du maximin share, mais il pourrait aussi être pertinent de vérifier si certains seuils de degré d'envie garantissent certaines propriétés.

Les travaux présentés dans ce chapitre, à l'exception de la réduction au problème de knapsack, ont été publiés à la conférence ECAI en même temps que les résultats du chapitre précédent (Gross-Humbert *et al.* [2023]).

Conclusion

Nous sommes partis du modèle de Singapour, un problème de *house allocation* ayant la particularité de partitionner l'ensemble des agents en types et l'ensemble des ressources en blocs, et d'imposer des quotas maximums entre ces types et blocs. Nous avons ensuite modifié ce modèle pour ajouter la notion d'homophilie aux préférences des agents. À l'aide de cette notion, nous avons défini une nouvelle dynamique en autorisant les agents à échanger leurs objets deux à deux s'ils préfèrent chacun l'objet de l'autre. L'introduction de cette dynamique a alors soulevé la question de la stabilité des allocations, dont nous avons étudié l'impact sur l'efficacité. Bien que cette analyse se soit révélée peu concluante, nous avons réussi à borner le prix de la stabilité entre 1 et 2. Nous souhaitons ensuite concevoir un mécanisme basé sur cette dynamique, mais pour cela il nous fallu en premier lieu discuter du mécanisme utilisé en situation réelle, afin de servir de point de comparaison. Nous avons donc expliqué le mécanisme de loterie et analysé les allocations qu'il est capable de produire, pour montrer que ces dernières n'auront aucune garantie d'efficacité ou de stabilité. Ceci fait, nous avons pu définir notre mécanisme d'échanges rationnels. Nous avons prouvé que ce mécanisme allait nécessairement terminer et retourner une allocation stable. Cependant, il ressort de son analyse qu'il n'a pas non plus de garantie d'efficacité particulière. Bien que reposant sur des principes très différents, les deux mécanismes présentent des résultats théoriques très semblables, nous nous sommes donc tournés vers des expériences pratiques pour les départager en profitant du fait qu'il existe un jeu de données réelles lié à l'immobilier de Singapour. Les résultats ont indiqué que si les performances des deux mécanismes en termes d'efficacité sont similaires, il est possible d'utiliser les deux successivement (mécanisme séquentiel puis mécanisme d'échanges) pour obtenir de meilleurs résultats. De plus, des expériences sur des données synthétiques et réelles montrent que si la complexité du mécanisme d'échanges n'est pas garantie d'être polynomiale, en pratique ce mécanisme finira rapidement.

L'étape suivante fut de chercher une notion d'équité adaptée à notre problème, avec comme objectif de trouver une manière de mesurer l'équité entre les types d'agents. Nous avons pour cela simplifié les instances de problèmes considérées (en retirant les notions de blocs et de quotas), et proposé une notion d'équité reposant sur les mêmes concepts que notre dynamique : l'envie. Cependant, notre modèle a des spécificités qui rendent la notion d'envie difficile à adapter : outre le fait que nos types puissent être de différentes tailles, les fonctions d'utilité des types ne sont pas additives.

Nous avons donc proposé un certain nombre d'axiomes intuitifs pour déterminer si une définition d'envie se comporte correctement, puis nous avons considéré des notions d'envie utilisées dans la littérature. Ces notions ayant été conçues pour des modèles différents, aucune n'était appropriée pour le nôtre : de fait, aucune notion considérée ne satisfaisait tous les axiomes. Nous avons donc conçu une nouvelle famille de notion d'envie, basée sur un même principe : plutôt que de comparer deux types entre eux, nous avons proposé de comparer le plus petit type à un ou plusieurs sous-groupes du plus grand type. L'une de ces définitions d'envie, l'envie pour au moins un sous-groupe, vérifie tous nos axiomes. Cependant, chercher des allocations sans-envie pour cette définition est difficile. De plus elle est trop contraignante pour que ce soit possible dans de nombreux cas. Par conséquent, nous avons dû chercher une relaxation.

Nous avons donc défini le degré d'envie : là où l'envie pour au moins un sous-groupe consiste à considérer si la comparaison résulte en envie pour au moins un des sous-groupes, le degré d'envie consiste à compter combien d'entre eux sont dans ce cas. Nous avons ensuite montré que cette mesure était également difficile à calculer de manière exacte, puisqu'elle appartenait à la classe de complexité $\#P$. Il nous a donc fallu chercher des méthodes d'approximation, et nous sommes parvenus à en trouver deux en nous restreignant au cas où les agents d'un même type partagent les mêmes utilités pour les objets. La première, plus simple, consiste à transformer une instance du problème du degré d'envie en une instance du problème du *#Exactly-M-items-knapsack*, pour lequel il existe déjà un algorithme d'approximation polynomial. La deuxième, plus complexe, repose sur une succession de marches aléatoires de Markov. Si cette dernière termine techniquement en temps polynomial, en pratique, les facteurs et exposants impliqués dans le calcul du nombre d'opérations sont si élevés que cette technique semble inutilisable en pratique. Or, cette méthode, contrairement à la première, peut être étendue naturellement au cas où les fonctions d'utilités sont individuelles, au prix de la borne théorique sur la complexité de calcul, elle présente donc un intérêt particulier malgré ses inconvénients. Nous avons donc implémenté cette méthode pour vérifier par l'expérience le nombre d'étapes nécessaires en pratique. Les expériences suggèrent que ce nombre est en pratique beaucoup plus petit que ce que la borne théorique laisse penser, et que considérer le cas des fonctions d'utilités individuelles n'altère pas la difficulté de calcul.

Perspectives

À partir de ces travaux, plusieurs perspectives d'études se dégagent.

La première revient à cette idée d'axiomatique exprimée dans le chapitre 3. Dans ce chapitre, nous souhaitons formaliser nos exigences concernant la notion d'envie que nous choisirons, nous avons donc établi nos 4 axiomes. La raison pour laquelle nous avons dû établir nos propres axiomes est qu'il existe très peu de littérature abordant les concepts d'équité par une approche axiomatique. Il n'existe de fait que quelques prémisses (Bosmans et Öztürk [2018], Thomson [2019]) de classification des différentes notions et mécanismes utilisés dans le domaine, mais ces derniers ne sont considérés

que individuellement et il n'existe pas d'étude générale cherchant à les classer en fonction de leurs propriétés. Nous avons donc dû concevoir nous-même des axiomes pour discriminer les différentes notions. Cependant, ces axiomes, bien qu'a priori pertinents pour la plupart des modèles, ne représentent qu'une petite partie de ce qui caractérise les notions d'envie considérées, et une plus petite partie encore de ce qui peut caractériser une mesure d'équité (ou, de manière plus générale, une mesure de qualité). Nous nous sommes de fait limité à ce qui nous était nécessaire pour discriminer les notions que nous avons considérées. Nous avançons donc qu'il existe un manque dans la littérature d'une étude cherchant à lister les principales mesures d'équité utilisées, déterminer leurs points communs et leurs différences, puis catégoriser ces mesures en fonction des propriétés respectées et enfreintes par chacune d'entre elles en fonction des modèles. Les objectifs d'une telle étude seraient multiples. Le premier serait la centralisation et la mise en relation des connaissances déjà existantes : il existe déjà des travaux considérant plusieurs notions et mettant en relation leurs résultats sur ces notions ; cependant, ces travaux considèrent en général un petit nombre de notions, et dans le cadre d'un seul modèle (ou d'un très petit nombre de modèles). Une étude plus étendue pourrait donc nous permettre un point de vue plus global sur le domaine. Un deuxième objectif serait d'introduire un formalisme pouvant servir de point de comparaison à partir duquel de futurs travaux pourraient plus facilement comparer une nouvelle mesure à celles déjà existantes, ou situer un modèle par rapport aux propriétés qui lui sont pertinentes. Enfin, un troisième objectif serait de détecter si d'éventuels résultats propres à une mesure particulière seraient généralisables à toute une classe de mesures partageant certaines propriétés.

Une autre perspective naturelle serait de chercher à adapter l'envie pour au moins un sous-groupe à un modèle disposant, en plus de types d'agents, de blocs d'objets et de quotas entre ces types et ces blocs. Si une telle modification paraît simple, elle soulève d'importantes questions sur les comparaisons autorisées dans le cadre de l'envie pour au moins un sous-groupe et celles que les quotas interdiront. En effet, si l'on souhaite mesurer l'envie entre deux types d'agents, ou même deux groupes quelconques d'agents, de tailles différentes dans un contexte où les blocs ont des quotas sur les types, il est nécessaire de décider si les agents sont autorisés ou non à considérer des objets que les quotas leur interdiraient de prendre. Cette question est plus compliquée qu'il n'y paraît en raison du grand nombre de configurations possibles et des particularités de l'envie pour au moins un sous-groupe (puisque des agents non impliqués dans la comparaison peuvent bloquer d'autres agents de leur type). Une manière d'étendre ce modèle sans multiplier les contraintes serait de considérer des "soft quotas", que les agents pourraient enfreindre dans certaines circonstances. Notre modèle se rapprocherait alors du problème du choix des écoles, mais resterait bien différent dans la mesure où les objets n'auraient pas de préférences sur les agents. Cela nous permettrait d'utiliser des concepts habituellement propres à ce problème, notamment les concepts d'envie "justifiée". Cette notion consiste à ne pas considérer les quotas comme des contraintes (d'où "soft quotas"), mais comme des indicateurs qui vont changer ce qui sera considéré comme une situation avec envie. Ainsi, nous pouvons faire en sorte qu'être en-

vieux d'un agent respectant les quotas requiert des circonstances très strictes en terme de quotas et d'utilité, là où être envieux d'un agent qui enfreint les quotas ne requiert que des circonstances courantes. Nous pouvons également décider que, lorsque l'on cherchera à minimiser l'envie, l'envie envers les agents qui enfreignent les contrats devra être considérée en priorité. L'objectif ici n'est pas seulement de définir une envie justifiée entre agents, mais de conjuguer cette définition avec celle de l'envie pour au moins un sous-groupe pour obtenir une envie justifiée "de groupe". Ce passage à l'échelle pose de nombreuses questions sur la manière dont seront traités les quotas, puisque là où les infractions sont binaires dans le cas individuel (un agent enfreint un quota ou non), elles peuvent être beaucoup plus complexes dans le cas d'un type au complet (combien de quotas sont enfreints, et de combien sont-ils enfreints ?). Une solution possible serait de pouvoir ordonner les infractions et de favoriser les envies lorsque le groupe envieux "enfreindraient moins" les quotas.

Une perspective qui se dégage de la précédente serait de concevoir une dynamique d'échanges entre agents adaptée à ce modèle. Dans le modèle classique, le lien entre envie individuelle et échanges rationnels est très simple : deux agents voudront échanger s'ils sont tout deux envieux l'un envers l'autre, l'échange va alors supprimer cette envie. En revanche, si l'un d'eux s'oppose à l'échange de ressource, alors forcer l'échange rendrait cet agent envieux de l'autre agent et serait donc contre-productif. Si on utilise plutôt la notion d'envie justifiée, il sera alors dans certaines circonstances possible de la réduire en forçant un échange où l'un des agents est récalcitrant. Imaginons un couple d'agent (a_1, a_2) où a_1 envie a_2 de manière justifiée mais a_2 ne souhaite pas échanger, si l'on force l'échange, alors l'envie de a_2 dans la nouvelle allocation ne sera pas justifiée. Nous pouvons donc essayer de définir une nouvelle dynamique inspirée de notre dynamique d'échanges qui se différenciera par le fait que nous autoriserons dans certains cas les agents à "forcer" un échange avec un autre agent. Bien sûr, il existe plusieurs obstacles à considérer dans la conception d'une telle dynamique. Le premier est le problème de la terminaison. La preuve de terminaison actuelle n'est plus applicable, mais une nouvelle preuve reposerait sans doute également sur un jeu de potentiel. En effet, si nous n'autorisons les échanges forcés que lorsque la quantité d'envie justifiée dans le système diminue, il est possible que cette quantité (ou du moins une valeur calculée à partir de cette quantité) n'augmente jamais, ce qui nous garantirait la terminaison. La deuxième difficulté serait liée au danger que cette dynamique introduit dans les échanges rationnels : en effet, il serait légitime pour un agent de refuser un échange améliorant, de crainte d'être par la suite sujet à un échange forcé lui donnant un bien de moindre qualité. De manière générale, la possibilité qu'une telle situation arrive semble intuitivement rendre une telle dynamique difficilement utilisable en situation réelle. Une manière d'éviter cela serait alors de définir une dynamique en deux temps, où l'on effectuerait en premier lieu les échanges forcés, puis les échanges rationnels. Une telle définition nécessite cependant d'avoir la certitude que les échanges rationnels ne risquent pas de permettre d'autres échanges forcés, ce qui peut requérir d'ajouter des contraintes sur les conditions dans lesquelles ces échanges peuvent avoir lieu. Cela nous empêchera au minimum de considérer la notion d'homophilie dans

ce modèle, puisqu'il est évident que si les échanges impactent les agents qui n'y participent pas, il sera possible pour un agent de spontanément changer son envie après un échange rationnel impliquant l'un de ses voisins. Une fois que notre nouvelle dynamique sera définie, nous pourrons nous consacrer à l'étude d'une nouvelle définition de la stabilité, ainsi qu'aux considérations d'efficacité et de complexité pratique.

Papers

Nathanaël GROSS-HUMBERT, Nawal BENABBOU, Aurélie BEYNIER et Nicolas MAUDET : Sequential and Swap Mechanisms for Public Housing Allocation with Quotas and Neighbourhood-Based Utilities (Extended Abstract). *In 20th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2021)*, London / Online, United Kingdom, mai 2021. URL <https://hal.science/hal-03210460>.

Nathanaël GROSS-HUMBERT, Nawal BENABBOU, Aurélie BEYNIER et Nicolas MAUDET : Sequential and Swap Mechanisms for Public Housing Allocation with Quotas and Neighbourhood-based Utilities. *ACM Transactions on Economics and Computation*, 10 (4), 2022. URL <https://hal.sorbonne-universite.fr/hal-03953844>.

Nathanaël GROSS-HUMBERT, Nawal BENABBOU, Aurélie BEYNIER et Nicolas MAUDET : On the Notion of Envy Among Groups of Agents in House Allocation Problems. *In 26th European Conference on Artificial Intelligence*, volume 372 de *Frontiers in Artificial Intelligence and Applications*, pages 924–931, Krakow, Poland, septembre 2023. IOS Press. URL <https://hal.science/hal-04362578>.

Bibliography

- Atila ABDULKADIROĞLU et Tayfun SÖNMEZ : Random serial dictatorship and the core from random endowments in house allocation problems. *Econometrica*, 66(3):689–701, 1998. ISSN 00129682, 14680262. URL <http://www.jstor.org/stable/2998580>. 8, 22, 50
- Rediet ABEBE, Jon M. KLEINBERG et David C. PARKES : Fair division via social comparison. In *Proc. 16th Conf. on Autonomous Agents and Multiagent Systems (AAMAS'17)*, pages 281–289, 2017. URL https://econcs.seas.harvard.edu/files/econcs/files/abebe_aamas17.pdf. 67
- Aishwarya AGARWAL, Edith ELKIND, Jiarui GAN et Alexandros A. VOUDOURIS : Swap stability in schelling games on graphs. In *Proceedings of the 34th AAAI Conference on Artificial Intelligence, AAAI 2020*, pages 1758–1765, 2020. URL <https://aaai.org/ojs/index.php/AAAI/article/view/5541>. 38
- Stéphane AIRIAU, Hugo GILBERT, Umberto GRANDI, Jérôme LANG et Anaëlle WILCZYNSKI : Fair rent division on a budget revisited. In *European Conference on Artificial Intelligence, 2023*. URL <https://api.semanticscholar.org/CorpusID:262219724>. 7
- Martin ALEKSANDROV et Toby WALSH : Group envy freeness and group pareto efficiency in fair division with indivisible items. In Frank TROLLMANN et Anni-Yasmin TURHAN, éditeurs : *KI 2018: Advances in Artificial Intelligence*, pages 57–72, Cham, 2018a. Springer International Publishing. ISBN 978-3-030-00111-7. 10, 69, 73, 77
- Martin ALEKSANDROV et Toby WALSH : Group envy freeness and group pareto efficiency in fair division with indivisible items. In Frank TROLLMANN et Anni-Yasmin TURHAN, éditeurs : *KI 2018: Advances in Artificial Intelligence*, pages 57–72, Cham, 2018b. Springer International Publishing. ISBN 978-3-030-00111-7. 30
- Haris AZIZ, Ioannis CARAGIANNIS, Ayumi IGARASHI et Toby WALSH : Fair allocation of indivisible goods and chores. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*, pages 53–59. International Joint Conferences on Artificial Intelligence Organization, 7 2019a. URL <https://doi.org/10.24963/ijcai.2019/8>. 12

- Haris AZIZ, Serge GASPERS et Zhaohong SUN : Mechanism design for school choice with soft diversity constraints. In Christian BESSIERE, éditeur : *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20*, pages 153–159. International Joint Conferences on Artificial Intelligence Organization, 7 2020. URL <https://doi.org/10.24963/ijcai.2020/22>. Main track. 23, 35
- Haris AZIZ, Serge GASPERS, Zhaohong SUN et Toby WALSH : From matching with diversity constraints to matching with regional quotas. In *Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems, AAMAS '19*, page 377–385, Richland, SC, 2019b. International Foundation for Autonomous Agents and Multiagent Systems. ISBN 9781450363099. 34
- Haris AZIZ et Simon REY : Almost group envy-free allocation of indivisible goods and chores. In Christian BESSIERE, éditeur : *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20*, pages 39–45. International Joint Conferences on Artificial Intelligence Organization, 7 2020. URL <https://doi.org/10.24963/ijcai.2020/6>. Main track. 7, 12, 29
- Haris AZIZ et Zhaohong SUN : School choice with flexible diversity goals and specialized seats. In Zhi-Hua ZHOU, éditeur : *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21*, pages 17–23. International Joint Conferences on Artificial Intelligence Organization, 8 2021. URL <https://doi.org/10.24963/ijcai.2021/3>. Main Track. 7, 21, 33
- Siddharth BARMAN, Arpita BISWAS, Sanath KRISHNAMURTHY et Yadati NARAHARI : Groupwise maximin fair allocation of indivisible goods. *Proceedings of the AAAI Conference on Artificial Intelligence*, 32(1), Apr. 2018. URL <https://ojs.aaai.org/index.php/AAAI/article/view/11463>. 28
- John BARTHOLDI et Michael A TRICK : Stable matching with preferences derived from a psychological model. *Operations Research Letters*, 5(4):165–169, 1986. ISSN 0167-6377. URL <https://www.sciencedirect.com/science/article/pii/0167637786900726>. 13
- Xiaohui BEI, Xinhang LU, Pasin MANURANGSI et Warut SUKSOMPONG : The price of fairness for indivisible goods. *Theory of Computing Systems*, 65(7):1069–1093, mars 2021. ISSN 1433-0490. URL <http://dx.doi.org/10.1007/s00224-021-10039-8>. 31
- Xiaohui BEI, Youming QIAO et Shengyu ZHANG : Networked fairness in cake cutting. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI-17*, pages 3632–3638, 2017. URL <https://doi.org/10.24963/ijcai.2017/508>. 67
- Nawal BENABBOU, Mithun CHAKRABORTY, Edith ELKIND et Yair ZICK : Fairness Towards Groups of Agents in the Allocation of Indivisible Items. In *The 28th In-*

- ternational Joint Conference on Artificial Intelligence, IJCAI'19*, 2019. URL <https://hal.sorbonne-universite.fr/hal-02155024>. 23, 25, 29
- Nawal BENABBOU, Mithun CHAKRABORTY, Xuan-Vinh Ho, Jakub SLIWINSKI et Yair ZICK : Diversity constraints in public housing allocation. *In Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems, AAMAS'18*, page 973–981, 2018. 8, 10, 23, 32, 33, 34, 37, 38, 41, 43, 50, 63, 69, 75
- Nawal BENABBOU, Mithun CHAKRABORTY, Ayumi IGARASHI et Yair ZICK : *Finding Fair and Efficient Allocations When Valuations Don't Add Up*, page 32–46. Springer International Publishing, 2020. ISBN 9783030579807. URL http://dx.doi.org/10.1007/978-3-030-57980-7_3. 72
- AuréliE BEYNIER, Sylvain BOUVERET, Michel LEMAÎTRE, Nicolas MAUDET, Simon REY et Parham SHAMS : Efficiency, sequenceability and deal-optimality in fair division of indivisible goods. *In Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems, AAMAS '19*, page 900–908, Richland, SC, 2019a. International Foundation for Autonomous Agents and Multiagent Systems. ISBN 9781450363099. 55
- AuréliE BEYNIER, Yann CHEVALEYRE, Laurent GOURVÈS, Ararat HARUTYUNYAN, Julien LESCA, Nicolas MAUDET et Anaëlle WILCZYNSKI : Local envy-freeness in house allocation problems. *Autonomous Agents and Multi-Agent Systems*, 33(5):591–627, 2019b. 67
- Anna BOGOMOLNAIA et Matthew O. JACKSON : The stability of hedonic coalition structures. *Games and Economic Behavior*, 38(2):201–230, 2002. ISSN 0899-8256. URL <https://www.sciencedirect.com/science/article/pii/S0899825601908772>. 15
- Kristof BOSMANS et Z. Emel ÖZTÜRK : An axiomatic approach to the measurement of envy. *Social Choice and Welfare*, 50:247–264, 2018. URL <https://api.semanticscholar.org/CorpusID:46872520>. 69, 144
- Sylvain BOUVERET, Yann CHEVALEYRE et Nicolas MAUDET : Fair allocation of indivisible goods. *In Felix BRANDT, Vincent CONITZER, Ulle ENDRISS, Jérôme LANG et Ariel D. PROCACCIA, éditeurs : Handbook of Computational Social Choice*, pages 284–310. Cambridge University Press, 2016. URL <https://doi.org/10.1017/CB09781107446984.013>. 7, 11
- Sylvain BOUVERET et Jérôme LANG : A general elicitation-free protocol for allocating indivisible goods. *In Toby WALSH, éditeur : IJCAI 2011, Proceedings of the 22nd International Joint Conference on Artificial Intelligence, Barcelona, Catalonia, Spain, July 16-22, 2011*, pages 73–78. IJCAI/AAAI, 2011. URL <https://doi.org/10.5591/978-1-57735-516-8/IJCAI11-024>. 50
- Sylvain BOUVERET et Michel LEMAÎTRE : Characterizing conflicts in fair division of indivisible goods using a scale of criteria. *Autonomous Agents and Multi-Agent Systems*,

- 30(2):259–290, 2015. ISSN 1573-7454. URL <http://recherche.noiraudes.net/resources/papers/JAAMAS15.pdf>. 29, 72
- Steven J. BRAMS et Alan D. TAYLOR : *Fair Division: From Cake-Cutting to Dispute Resolution*. Cambridge University Press, 1996. 7, 18, 31
- Robert BREDERECK, Edith ELKIND et Ayumi IGARASHI : Hedonic diversity games. *In Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems, AAMAS'19*, page 565–573, 2019. ISBN 9781450363099. 9, 15, 16
- Robert BREDERECK, Andrzej KACZMARCZYK et Rolf NIEDERMEIER : Envy-free allocations respecting social networks. *Artificial Intelligence*, 305:103664, 2022. ISSN 0004-3702. URL <https://www.sciencedirect.com/science/article/pii/S0004370222000042>. 67
- Xiaolin BU, Jiaxin SONG et Ziqi YU : Efx allocations exist for binary valuations. *In Frontiers of Algorithmics: 17th International Joint Conference, IJTCS-FAW 2023 Macau, China, August 14–18, 2023 Proceedings*, page 252–262, Berlin, Heidelberg, 2023. Springer-Verlag. ISBN 978-3-031-39343-3. URL https://doi.org/10.1007/978-3-031-39344-0_19. 10, 29
- Eric BUDISH : The combinatorial assignment problem: Approximate competitive equilibrium from equal incomes. *Journal of Political Economy*, 119(6):1061 – 1103, 2011. URL <https://EconPapers.repec.org/RePEc:ucp:jpolec:doi:10.1086/664613>. 10, 27, 28, 72, 115
- Franklin CAMACHO, Rigoberto FONSECA-DELGADO, Ramón PINO PÉREZ et Guido TAPIA : Generalized binary utility functions and fair allocations. *Mathematical Social Sciences*, 121:50–60, 2023. ISSN 0165-4896. URL <https://www.sciencedirect.com/science/article/pii/S0165489622000798>. 31
- Ioannis CARAGIANNIS, David KUOKAWA, Hervé MOULIN, Ariel D. PROCACCIA, Nisarg SHAH et Junxing WANG : The unreasonable fairness of maximum nash welfare. *ACM Trans. Econ. Comput.*, 7(3), sep 2019. ISSN 2167-8375. URL <https://doi.org/10.1145/3355902>. 10, 12, 25, 27, 28, 31, 72
- Mithun CHAKRABORTY, Ayumi IGARASHI, Warut SUKSOMPONG et Yair ZICK : Weighted envy-freeness in indivisible item allocation. *ACM Trans. Econ. Comput.*, 9(3), aug 2021. ISSN 2167-8375. URL <https://doi.org/10.1145/3457166>. 10, 76
- Adhip CHAUDHURI : Some implications of an intensity measure of envy. *Social Choice and Welfare*, 3:255–270, 1986. URL <https://api.semanticscholar.org/CorpusID:153957928>. 115
- Bhaskar Ray CHAUDHURY, Jugal GARG et Kurt MEHLHORN : Efx exists for three agents. *J. ACM*, 71(1), feb 2024. ISSN 0004-5411. URL <https://doi.org/10.1145/3616009>. 29

- Ankit CHAUHAN, Pascal LENZNER et Louise MOLITOR : Schelling segregation with strategic agents. In *Algorithmic Game Theory: 11th International Symposium, SAGT 2018, Beijing, China, September 11-14, 2018, Proceedings*, page 137–149, Berlin, Heidelberg, 2018. Springer-Verlag. ISBN 978-3-319-99659-2. URL https://doi.org/10.1007/978-3-319-99660-8_13. 9, 17, 38, 41, 55
- Chandra R. CHEGIREDDY et Horst W. HAMACHER : Algorithms for finding k -best perfect matchings. *Discrete Applied Mathematics*, 18(2):155–165, 1987. ISSN 0166-218X. URL <https://www.sciencedirect.com/science/article/pii/0166218X87900175>. 96, 124
- Yann CHEVALEYRE, Ulrich ENDRISS et Nicolas MAUDET : Simple negotiation schemes for agents with simple preferences: sufficiency, necessity and maximality. *Autonomous Agents and Multi-Agent Systems*, 20:234–259, 2010. URL <https://api.semanticscholar.org/CorpusID:18729123>. 46
- Vincent CONITZER, Rupert FREEMAN, Nisarg SHAH et Jennifer Wortman VAUGHAN : Group fairness for the allocation of indivisible goods. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33(01):1853–1860, Jul. 2019. URL <https://ojs.aaai.org/index.php/AAAI/article/view/4010>. 25, 73
- Anastasia DAMAMME, Aurélie BEYNIER, Yann CHEVALEYRE et Nicolas MAUDET : The Power of Swap Deals in Distributed Resource Allocation. In *The 14th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2015)*, pages 625–633, Istanbul, Turkey, mai 2015. URL <https://hal.science/hal-01212603>. 9, 32, 46, 67
- Dimitrios DIAMANTARAS et William THOMSON : A refinement and extension of the no-envy concept. *Economics Letters*, 33(3):217–222, 1990. ISSN 0165-1765. URL <https://www.sciencedirect.com/science/article/pii/016517659090004K>. 115
- Jacques H. DREZE : Market allocation under uncertainty. *European Economic Review*, 2(2):133–165, 1970. ISSN 0014-2921. URL <https://www.sciencedirect.com/science/article/pii/0014292170900103>. 26
- J. H. DRÈZE et J. GREENBERG : Hedonic coalitions: Optimality and stability. *Econometrica*, 48(4):987–1003, 1980. ISSN 00129682, 14680262. URL <http://www.jstor.org/stable/1912943>. 15
- Federico ECHENIQUE et M. Bumin YENMEZ : How to control controlled school choice. *American Economic Review*, 105(8):2679–94, August 2015. URL <https://www.aeaweb.org/articles?id=10.1257/aer.20130929>. 8, 23, 33
- Lars EHLERS, Isa E. HAFALIR, M. Bumin YENMEZ et Muhammed A. YILDIRIM : School choice with controlled choice constraints: Hard bounds versus soft bounds. *Journal of Economic Theory*, 153:648–683, 2014. ISSN 0022-0531. URL <https://www.sciencedirect.com/science/article/pii/S0022053114000301>. 34

- Edith ELKIND, Jiarui GAN, Ayumi IGARASHI, Warut SUKSOMPONG et Alexandros A. VOUDOURIS : Schelling games on graphs. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI'19*, pages 266–272, 7 2019. URL <https://doi.org/10.24963/ijcai.2019/38>. 17, 38, 42
- Edith ELKIND, Neel PATEL, Alan TSANG et Yair ZICK : Keeping your friends close: Land allocation with friends. In Christian BESSIERE, éditeur : *Proceedings of the 29th International Joint Conference on Artificial Intelligence, IJCAI'20*, pages 318–324, 7 2020. URL <https://doi.org/10.24963/ijcai.2020/45>. 38, 41
- Peter EMERSON : The original borda count and partial voting. *Social Choice and Welfare*, 40:353–358, 2013. URL <https://api.semanticscholar.org/CorpusID:29826994>. 23
- U. ENDRISS, N. MAUDET, F. SADRI et F. TONI : Negotiating socially optimal allocations of resources. *Journal of Artificial Intelligence Research*, 25:315–348, mars 2006. ISSN 1076-9757. URL <http://dx.doi.org/10.1613/jair.1870>. 9
- Allan M. FELDMAN et Alan KIRMAN : Fairness and envy. *The American Economic Review*, 64:995–1005, 1974. URL <https://api.semanticscholar.org/CorpusID:152436138>. 115
- Duncan K. FOLEY : Resource allocation and the public sector. *Yale Economic Essays*, 7 (1):45–98, 1967. 7, 9, 28, 69, 72
- Thibault GAJDOS et Jean-Marc TALLON : Fairness under uncertainty. *Economics Bulletin*, 4:1–7, 2002. URL <https://api.semanticscholar.org/CorpusID:54845850>. 29
- Ya'akov (Kobi) GAL, Moshe MASH, Ariel D. PROCACCIA et Yair ZICK : Which is the fairest (rent division) of them all? In *Proceedings of the 2016 ACM Conference on Economics and Computation, EC '16*, page 67–84, New York, NY, USA, 2016. Association for Computing Machinery. ISBN 9781450339360. URL <https://doi.org/10.1145/2940716.2940724>. 19
- D. GALE et L. S. SHAPLEY : College admissions and the stability of marriage. *The American Mathematical Monthly*, 69(1):9–15, 1962. URL <https://doi.org/10.1080/00029890.1962.11989827>. 7, 13, 15, 21
- Michael R. GAREY et David S. JOHNSON : *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman & Co., USA, 1979. ISBN 0716710447. 119
- Mohammad GHODSI, Mohammadtaghi HAJIAGHAYI, Masoud SEDDIGHIN, Saeed SEDDIGHIN et Hadi YAMI : Fair allocation of indivisible goods: Improvements and generalizations. In *Proceedings of the 2018 ACM Conference on Economics and Computation, EC '18*, page 539–556, New York, NY, USA, 2018. Association for Computing Machinery. ISBN 9781450358293. URL <https://doi.org/10.1145/3219166.3219238>. 30

- Masahiro GOTO, Fuhiko KOJIMA, Ryoji KURATA, Akihisa TAMURA et Makoto YOKOO : Designing matching mechanisms under general distributional constraints. *In Proceedings of the Sixteenth ACM Conference on Economics and Computation, EC '15*, page 259–260, New York, NY, USA, 2015. Association for Computing Machinery. ISBN 9781450334105. URL <https://doi.org/10.1145/2764468.2764501>. 21
- Laurent GOURVÈS, Jérôme MONNOT et Lydia TLILANE : Near fairness in matroids. *In European Conference on Artificial Intelligence*, 2014. URL <https://api.semanticscholar.org/CorpusID:326583>. 28
- Venkatesan GURUSWAMI : Rapidly mixing markov chains: A comparison of techniques (a survey). *ArXiv*, abs/1603.01512, 2016. URL <https://api.semanticscholar.org/CorpusID:6448773>. 128, 129, 130
- Daniel HALPERN, Ariel D. PROCACCIA, Alexandros PSOMAS et Nisarg SHAH : Fair division with binary valuations: One rule to rule them all. *In Web and Internet Economics: 16th International Conference, WINE 2020, Beijing, China, December 7–11, 2020, Proceedings*, page 370–383, Berlin, Heidelberg, 2020. Springer-Verlag. ISBN 978-3-030-64945-6. URL https://doi.org/10.1007/978-3-030-64946-3_26. 10, 25, 31
- Mark JERRUM et Alistair SINCLAIR : The markov chain monte carlo method: An approach to approximate counting and integration. *Approximation Algorithms for NP-Hard Problems*, page 482–520, 1996. 122
- Panagiotis KANELLOPOULOS, Maria KYROPOULOU et Alexandros A. VOUDOURIS : Modified schelling games. *In Algorithmic Game Theory: 13th International Symposium, SAGT 2020, Augsburg, Germany, September 16–18, 2020, Proceedings*, page 241–256, Berlin, Heidelberg, 2020. Springer-Verlag. ISBN 978-3-030-57979-1. URL https://doi.org/10.1007/978-3-030-57980-7_16. 38, 42
- Richard M. KARP : *Reducibility among Combinatorial Problems*, pages 85–103. Springer US, Boston, MA, 1972. ISBN 978-1-4684-2001-2. URL https://doi.org/10.1007/978-1-4684-2001-2_9. 119
- David A. KOHLER et Ramaswamy CHANDRASEKARAN : A class of sequential games. *Oper. Res.*, 19:270–277, 1971. URL <https://api.semanticscholar.org/CorpusID:207238185>. 32, 50
- Ryoji KURATA, Masahiro GOTO, Atsushi IWASAKI et M. YOKOO : Controlled school choice with soft bounds and overlapping types. *J. Artif. Intell. Res.*, 58:153–184, 2015. URL <https://api.semanticscholar.org/CorpusID:1207566>. 34, 35
- Maria KYROPOULOU, Warut SUKSOMPONG et Alexandros A. VOUDOURIS : Almost envy-freeness in group resource allocation. *Theoretical Computer Science*, 841:110–123, 2020. ISSN 0304-3975. URL <https://www.sciencedirect.com/science/article/pii/S0304397520303807>. 19

- R. J. LIPTON, E. MARKAKIS, E. MOSSEL et A. SABERI : On approximately fair allocations of indivisible goods. *In Proceedings of the 5th ACM Conference on Electronic Commerce, EC '04*, page 125–131, New York, NY, USA, 2004. Association for Computing Machinery. ISBN 1581137710. URL <https://doi.org/10.1145/988772.988792>. 30
- Pasin MANURANGSI et Warut SUKSOMPONG : Almost envy-freeness for groups: Improved bounds via discrepancy theory. *Theoretical Computer Science*, 930:179–195, 2022. ISSN 0304-3975. URL <https://www.sciencedirect.com/science/article/pii/S0304397522004546>. 115
- Hervi MOULIN : *Axioms of Cooperative Decision Making*. Econometric Society Monographs. Cambridge University Press, 1988. 23
- Hervé MOULIN et William THOMSON : Can everyone benefit from growth?: Two difficulties. *Journal of Mathematical Economics*, 17(4):339–345, 1988. ISSN 0304-4068. URL <https://www.sciencedirect.com/science/article/pii/030440688890016X>. 103, 110
- James E. NEWMAN et Moshe Y. VARDI : Fpras approximation of the matrix permanent in practice. *ArXiv*, abs/2012.03367, 2020. URL <https://api.semanticscholar.org/CorpusID:227344085>. 138, 139
- Benjamin PLAUT et Tim ROUGHGARDEN : Almost envy-freeness with general valuations. *In Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA '18*, page 2584–2603, USA, 2018. Society for Industrial and Applied Mathematics. ISBN 9781611975031. 12, 29, 30, 31, 72
- RONALD DE HAAN : Why a Dutch court stopped high school students from swapping schools. <https://medium.com/social-choice/why-a-dutch-court-stopped-high-school-students-from-exchanging-schools-1315303a48b6>, 2017. Accessed: 2020-10-08. 9
- Alvin E. ROTH et Marilda A. Oliveira SOTOMAYOR : *Two-Sided Matching: A Study in Game-Theoretic Modeling and Analysis*. Econometric Society Monographs. Cambridge University Press, 1990. 19
- Govind S. SANKAR, Anand LOUIS, Meghana NASRE et Prajakta NIMBHORKAR : Online algorithms for matchings with proportional fairness constraints and diversity constraints. *In European Conference on Artificial Intelligence*, 2022. URL <https://api.semanticscholar.org/CorpusID:251765097>. 34
- Jonathan SCARLETT, Nicholas TEH et Yair ZICK : For one and all: Individual and group fairness in the allocation of indivisible goods. *In Proceedings of the 2023 International Conference on Autonomous Agents and Multiagent Systems, AAMAS '23*, page 2466–2468, Richland, SC, 2023. International Foundation for Autonomous Agents and Multiagent Systems. ISBN 9781450394321. 10, 69, 79

- Thomas C. SCHELLING : Models of segregation. *The American Economic Review*, 59 (2):488–493, 1969. ISSN 00028282. URL <http://www.jstor.org/stable/1823701>. 9, 16, 37
- Erel SEGAL-HALEVI et Warut SUKSOMPONG : Democratic fair allocation of indivisible goods. *Artificial Intelligence*, 277:103167, 2019. ISSN 0004-3702. URL <https://www.sciencedirect.com/science/article/pii/S0004370218307082>. 19, 29
- Lloyd S. SHAPLEY et Herbert E. SCARF : On cores and indivisibility. *Journal of Mathematical Economics*, 1:23–37, 1974. URL <https://api.semanticscholar.org/CorpusID:154744803>. 46
- H. STEINHAUS : The problem of fair division. *Econometrica*, 16:101–104, 1948. URL <https://cir.nii.ac.jp/crid/1572261550519680768>. 7, 27
- William THOMSON : The fair division of a fixed supply among a growing population. *Math. Oper. Res.*, 8:319–326, 1983. URL <https://api.semanticscholar.org/CorpusID:14373861>. 103, 110
- William THOMSON : Axiomatic and game-theoretic analysis of bankruptcy and taxation problems: a survey. *Mathematical Social Sciences*, 45(3):249–297, 2003. ISSN 0165-4896. URL <https://www.sciencedirect.com/science/article/pii/S0165489602000707>. 18
- William THOMSON : *On the Axiomatics of Resource Allocation: Classifying Axioms and Mapping Out Promising Directions*, pages 213–222. Springer International Publishing, Cham, 2019. ISBN 978-3-030-18050-8. URL https://doi.org/10.1007/978-3-030-18050-8_29. 70, 144
- Theofilos TRIOMMATIS et Aris PAGOURTZIS : Approximate #knapsack computations to count semi-fair allocations. In Jianer CHEN, Qilong FENG et Jinhui XU, éditeurs : *Theory and Applications of Models of Computation*, pages 239–250, Cham, 2020. Springer International Publishing. ISBN 978-3-030-59267-7. 120, 121, 122
- U.S. DEPARTMENT OF EDUCATION, OFFICE OF ELEMENTARY AND SECONDARY EDUCATION : Improving outcomes for all students: Strategies and considerations to increase student diversity, 2017. URL <https://school-diversity.org/pdf/improving-outcomes-diversity.pdf>. 33
- Leslie G. VALIANT : The complexity of enumeration and reliability problems. *SIAM J. Comput.*, 8(3):410–421, aug 1979a. ISSN 0097-5397. URL <https://doi.org/10.1137/0208032>. 119
- L.G. VALIANT : The complexity of computing the permanent. *Theoretical Computer Science*, 8(2):189–201, 1979b. ISSN 0304-3975. URL <https://www.sciencedirect.com/science/article/pii/0304397579900446>. 118

Lin Zhou : On a conjecture by gale about one-sided matching problems. *Journal of Economic Theory*, 52(1):123–135, 1990. ISSN 0022-0531. URL <https://www.sciencedirect.com/science/article/pii/002205319090070Z>. 8, 22

Appendix A

Construction des permutations pour la preuve de la proposition 2.5.2

Nous allons maintenant expliquer comment créer une famille de permutations $(\sigma_p)_{p \in \llbracket 1, k \rrbracket}$ vérifiant les propriétés suivantes :

- Pour tout $p \in \llbracket 1, k \rrbracket$, σ_p est une permutation sur $\{1, \dots, k\} \setminus \{p\}$
- $\forall p \in \llbracket 1, k \rrbracket, q \in \llbracket 1, k \rrbracket \setminus \{p\}, \sigma_p(q) \neq q$
- $\forall p, p' \in \llbracket 1, k \rrbracket$ tels que $p \neq p', \forall q \in \llbracket 1, k \rrbracket \setminus \{p\}, \sigma_p(q) \neq \sigma_{p'}(q)$

Plutôt que de construire des permutations sur $\{1, \dots, k\} \setminus \{p\}$, nous allons construire des permutations sur $\{1, \dots, k\}$ en imposant que $\forall p \in \llbracket 1, k \rrbracket, \sigma_p(p) = p$.

Nous considérerons deux cas différents: k pair ou impair.

A.1 Cas k impair

Pour tout $p \in \llbracket 1, k-1 \rrbracket$, nous définissons la permutation σ_p sur $\llbracket 1, k \rrbracket$ par:

$$\sigma_p(q) = \begin{cases} p & \text{si } q - p \equiv 0 \pmod{k} \\ k & \text{si } q - p \equiv p \pmod{k} \\ q - p \pmod{k} & \text{sinon} \end{cases}$$

Notons tout d'abord que chaque σ_p est bien une permutation, puisque nous allons simplement parcourir les $q - p \pmod{k}$ et échanger les deux termes pour $q - p \equiv 0 \pmod{k}$ et $q - p \equiv p \pmod{k}$. Puisque p vaut entre 1 et $k-1$, nous aurons nécessairement $\forall q \neq p, \sigma_p(q) \neq q$ sauf en p . Il n'est de plus pas possible que le cas $\sigma_p(q) = k$ arrive pour $q = k$ car k est impair (si il était pair, il arriverait pour $p = 2k$ et $q = k$).

Il est de plus facile de vérifier que pour tout p, p' dans $\llbracket 1, k-1 \rrbracket$, si $p \neq p'$ alors $\sigma_p(q) \neq \sigma_{p'}(q)$ pour tout q .

Il nous reste à construire σ_k . Nous le définissons en posant:

$$\sigma_k(q) = \begin{cases} k & \text{si } q = k \\ q/2 & \text{si } q \equiv 0 \pmod{2} \\ (q+k)/2 & \text{sinon} \end{cases}$$

Notons en premier lieu que puisque k est impair, alors soit q est pair soit $q+k$ l'est. Il est de plus évident que σ_k est bien une permutation sur $[[1, k]]$ puisqu'il s'agit d'une injection de $[[1, k]]$ dans $[[1, k]]$.

Notons également que pour tout $q \in [[1, k-1]]$, nous aurons $\sigma_k(q) = p$ avec p tel que $\sigma_p(q) = k$. Cette propriété garantie que σ_k vérifie bien $\forall p < k, \forall q \in [[1, k]], \sigma_k(q) \neq \sigma_p(q)$ puisque nous savons que pour tout $p' \neq \sigma_k(q), q - p' \not\equiv p' \pmod{k}$.

A.2 Cas k pair

Dans ce cas, nous avons $k = 2k'$ avec $k' \in \mathbb{N}$.

Cette partie se comprend plus facilement si l'on conçoit la famille des permutations comme une matrice Σ carrée de taille k telle que $\Sigma_{p,q} = \sigma_p q$.

Nous allons commencer par définir les permutations σ_p pour $p \in [[2, k-1]]$ par blocs de 2 sur 2. Par conséquent, nous allons utiliser des indices i et j , qui parcourront $[[1, k'-1]] \times [[1, k']]$, pour dénoter les positions des bloc. Les quatre "cases" du bloc en position (i, j) seront alors $\sigma_{2i}(2*j-1), \sigma_{2i+1}(2*j), \sigma_{2i}(2*j)$ et $\sigma_{2i+1}(2*j-1)$.

La première étape sera de partitionner $S = [[1, k'-1]] \times [[1, k']]$ en deux ensembles S_1 et S_2 de taille égale. La partition dépendra de si k' ou $k'-1$ est pair:

- si k' est pair: $S_1 = [[1, k'-1]] \times [[1, k'/2]]$ et $S_2 = [[1, k'-1]] \times [[k'/2+1, k']]$.
- si $k'-1$ est pair: $S_1 = [[1, (k'-1)/2]] \times [[1, k']]$ et $S_2 = [[(k'-1)/2, k'-1]] \times [[1, k']]$

Il s'agit de deux manières de découper la matrice listant les permutations. Dans la pratique, ce découpage est seulement nécessaire pour s'assurer du bon traitement de la diagonale: nous passons de S_1 à S_2 quand nous entrons dans la deuxième moitié de la diagonale (c'est-à-dire les indices supérieurs à k').

Nous pouvons maintenant poser:

- Pour tout $(i, j) \in S_1$:

$$\sigma_{2i}(2*j-1) = \sigma_{2i+1}(2*j) = \begin{cases} k & \text{si } i+j \equiv 0 \pmod{k'} \\ (i+j \pmod{k'}) + k' & \text{sinon} \end{cases}$$

$$\sigma_{2i}(2*j) = \sigma_{2i+1}(2*j-1) = \begin{cases} k' & \text{si } i+j \equiv 0 \pmod{k'} \\ i+j \pmod{k'} & \text{sinon} \end{cases}$$

- Pour tout $(i, j) \in S_2$:

$$\sigma_{2i}(2*j-1) = \sigma_{2i+1}(2*j) = \begin{cases} k' & \text{si } i+j \equiv 0 \pmod{k'} \\ i+j \pmod{k'} & \text{sinon} \end{cases}$$

$$\sigma_{2i}(2*j) = \sigma_{2i+1}(2*j-1) = \begin{cases} k & \text{si } i+j \equiv 0 \pmod{k'} \\ (i+j \pmod{k'}) + k' & \text{sinon} \end{cases}$$

Plusieurs remarques:

- Chaque σ_p sera bien une permutation: à i constant, chaque $i+j$ et $i+j+k'$ sera bien différent. La même chose est vrai pour j , ce qui nous donne que $p \neq p' \implies \forall q, \sigma_p(q) \neq \sigma_{p'}(q)$.
- La diagonale sera toujours rencontrée dans un coin supérieur droite ($\sigma_{2i}(2*j)$) pour $i = j$, ou inférieur gauche ($\sigma_{2i+1}(2*j-1)$) pour $i+1 = j$. Il est facile de vérifier que nous aurons bien $\sigma_p(p) = p$ dans S_1 . Pour S_2 , cela vient du fait que sur la diagonale nous aurons $k' < i+j \leq k$, la disjonction de cas revient donc en fait à $i+j$.
- Conséquence des deux remarques précédentes, nous aurons bien $\forall q, \sigma_p(q) \neq q$ puisque $\sigma_p(q) \neq \sigma_q(q) = q$

Il nous reste maintenant à définir σ_1 et σ_k . Nous utiliserons la même logique, étendue entre la première et la dernière ligne, cependant il faut aussi inverser le dernier "bloc". Pour tout $j \in \llbracket 1, k'-1 \rrbracket$:

$$\begin{aligned} \sigma_1(2*j-1) &= \sigma_k(2*j) = j \\ \sigma_1(2*j) &= \sigma_k(2*j-1) = j+k' \end{aligned}$$

Et pour finir:

$$\begin{aligned} \sigma_1(k-1) &= \sigma_k(k) = k \\ \sigma_1(k) &= \sigma_k(k-1) = k' \end{aligned}$$

Appendix B

Preuve de la proposition 2.3.2

Nous allons maintenant prouver les bornes sur le prix de la stabilité PoS .

B.1 Preuve que $1.6 \leq PoS$

Nous allons prouver cette borne en construisant une instance qui l'atteint.

Nous considérons une instance I contenant un ensemble de 16 agents partitionné en 4 types: $T_1 = \{a_{1,1}, a_{1,2}, a_{1,3}, a_{1,4}\}$, $T_2 = \{a_{2,1}, a_{2,2}, a_{2,3}, a_{2,4}\}$, $T_3 = \{a_{3,1}, a_{3,2}, a_{3,3}, a_{3,4}\}$, et $T_4 = \{a_{4,1}, a_{4,2}, a_{4,3}, a_{4,4}\}$. De même, l'instance contient un ensemble de 16 objets partitionné en 4 blocs: $B_1 = \{h_{1,1}, h_{1,2}, h_{1,3}, h_{1,4}\}$, $B_2 = \{h_{2,1}, h_{2,2}, h_{2,3}, h_{2,4}\}$, $B_3 = \{h_{3,1}, h_{3,2}, h_{3,3}, h_{3,4}\}$, et $B_4 = \{h_{4,1}, h_{4,2}, h_{4,3}, h_{4,4}\}$.

Nous définissons les quotas de la manière suivante: $\lambda_{p,q} = 4$ si $p = q$, $\lambda_{p,q} = 1$ sinon. Nous posons $\varphi = 1$ et nous définissons les utilités de position en utilisant la matrice P_ϵ suivante (pour tout $\epsilon > 0$):

$$\forall i \in T_p, h \in B_q, u_i^I(h) = P_{p,q} \text{ où } P_\epsilon = \begin{pmatrix} 0 & \frac{3}{4} + \epsilon & \frac{2}{4} + \epsilon & \frac{1}{4} + \epsilon \\ \frac{3}{4} + \epsilon & 0 & \frac{1}{4} + \epsilon & \frac{2}{4} + \epsilon \\ \frac{2}{4} + \epsilon & \frac{1}{4} + \epsilon & 0 & \frac{3}{4} + \epsilon \\ \frac{1}{4} + \epsilon & \frac{2}{4} + \epsilon & \frac{3}{4} + \epsilon & 0 \end{pmatrix}$$

Trouver les allocations optimales et les allocations stables peut être fait en énumérant toutes les allocations valides ou en utilisant le programme linéaire en nombre entier présenté au chapitre 2.

Nous présentons ici seulement les résultats:

- Les allocations qui maximise le bien-être social utilitariste sont celles qui allouent tous les agents de chaque type T_p au bloc B_p ($p \in \{1, \dots, 4\}$). L'utilité de chaque agent est alors de $0 + \varphi \times \frac{4}{4} = 1$, le bien-être social utilitariste est donc de 16.

Une telle allocation n'est cependant pas stable, par exemple les agents de T_1 et de T_2 souhaitent échanger leur biens pour faire passer leur utilité de 1 à $1 + \epsilon$.

- Toute allocation allouant un seul agent de chaque type dans chaque bloc est une allocation valide et stable.

Le bien-être social utilitariste est alors de $4 \times (0 + \varphi \times \frac{1}{4}) + 4 \times (\frac{1}{4} + \varepsilon + \varphi \times \frac{1}{4}) + 4 \times (\frac{2}{4} + \varepsilon + \varphi \times \frac{1}{4}) + 4 \times (\frac{3}{4} + \varepsilon + \varphi \times \frac{1}{4}) = 10 + 12\varepsilon$.

Par conséquent, le ratio entre l'utilité d'une allocation optimale et celle de la meilleur allocation stable est majoré par $\frac{16}{10+12\varepsilon}$, qui tend vers 1.6 quand ε tends vers 0.

B.2 Preuve que $PoS \leq 2$

Considérons une instance quelconque parmi les instances pour lesquelles aucune allocation optimale n'est stable. Considérons ensuite l'une de ses allocations d'utilité maximale OPT . À partir de cette allocation, nous appliquons le mécanisme d'échanges rationnels pour atteindre l'allocation stable $POST$.

Notons que la séquence exacte des échanges effectués n'entre pas en considération, l'important est que $\frac{sw(OPT)}{sw(POST)}$ nous permette de borner le prix de la stabilité puisque l'utilité de $POST$ est inférieure ou égale à celle de la meilleur allocation stable. De plus, nous utiliserons le fait que $POST$ soit une allocation atteignable depuis OPT par des échanges rationnels.

Nous allons tout d'abord décomposer $sw(OPT)$ et $sw(POST)$ de la manière suivante:

$$\begin{aligned} sw(OPT) &= U_{OPT}^I + \varphi U_{OPT}^N \\ sw(POST) &= U_{POST}^I + \varphi U_{POST}^N \end{aligned}$$

où U_{OPT}^I (resp. U_{POST}^I) est la somme des utilité de position des agents dans l'allocation OPT (resp. $POST$), et U_{OPT}^N (resp. U_{POST}^N) est la somme des utilités de voisinage des agents dans l'allocation OPT (resp. $POST$).

Nous allons maintenant utiliser la fonction de potentielle pot introduite dans la preuve de la proposition 2.5.1. Rappelons que pot est définie par:

$$pot(A) = sw(A) + \sum_{i \in \mathcal{N}} u_i^I(A)$$

Dans notre cas, nous avons donc $pot(OPT) = sw(OPT) + U_{OPT}^I$ et $pot(POST) = sw(POST) + U_{POST}^I$.

Nous avons déjà prouvé dans la preuve de la proposition 2.5.1 que cette fonction augmente strictement à chaque échange rationnel. Or puisque OPT n'est pas stable (par hypothèse), au moins un échange a été effectué pour atteindre $POST$.

Par conséquent $pot(POST) > pot(OPT)$, c'est-à-dire $sw(POST) + U_{POST}^I > sw(OPT) + U_{OPT}^I$. Cependant, OPT est optimale, par conséquent $sw(OPT) > sw(POST)$. Nous avons donc nécessairement $U_{OPT}^I < U_{POST}^I$.

Nous avons donc

$$\frac{sw(OPT)}{sw(POST)} = \frac{sw(OPT) + sw(POST) - sw(POST)}{sw(POST)} = 1 + \frac{sw(OPT) - sw(POST)}{sw(POST)}$$

Puisque $sw(OPT) = pot(OPT) - U_{OPT}^I$ et $sw(POST) = pot(POST) - U_{POST}^I$, nous obtenons:

$$\begin{aligned} \frac{sw(OPT)}{sw(POST)} &= 1 + \frac{pot(OPT) - U_{OPT}^I - pot(POST) + U_{POST}^I}{sw(POST)} \\ &< 1 + \frac{U_{POST}^I - U_{OPT}^I}{sw(POST)} \quad (\text{puisque } pot(OPT) < pot(POST)) \\ &< 1 + \frac{U_{POST}^I}{sw(POST)} \quad (\text{puisque } U_{OPT}^I \geq 0) \\ &< 2 \quad (\text{puisque } sw(POST) = U_{POST}^I + \varphi U_{POST}^N) \end{aligned}$$

Puisque le ratio entre l'utilité d'une allocation optimale et l'utilité d'une allocation stable est inférieur à 2, nous pouvons déduire que le ratio entre l'utilité d'une allocation optimale et l'utilité de la meilleure allocation stable est inférieur 2.

Puisque le prix de la stabilité est la plus petite valeur majorant ce ratio sur toutes les instances possibles, nous en déduisons que $PoS \leq 2$.