

Partage équitable multiniveaux avec des préférences de type rang de matroïde*

¹ LIP6, Sorbonne Université

Maxime Lucet¹, Nawal Benabbou¹, Aurélie Beynier¹, Nicolas Maudet¹

{prénom.nom}@lip6.fr

Abstract

Nous introduisons un problème d'allocation équitable de ressources multiniveaux avec des relations hiérarchiques arborescentes entre les agents. À chaque niveau, le problème peut être considéré localement comme la répartition du lot de ressources d'un agent entre ses enfants, l'allocation globale étant la trace de ce processus itéré jusqu'aux feuilles. Chaque nœud interne peut ainsi potentiellement posséder son propre mécanisme d'allocation local. Nous proposons deux algorithmes en supposant des utilités de rang de matroïde au niveau des feuilles et des utilités additives pour les nœuds internes : un algorithme séquentiel descendant en temps polynomial, générique (compatible avec divers algorithmes locaux) et offrant des garanties théoriques d'efficacité et d'équité; ainsi qu'une extension du General Yankee Swap adaptée à ce cadre, qui s'exécute en temps polynomial, garantit l'efficacité et fait preuve d'une forte équité empirique en pratique.

Keywords

Fair allocation, computational social choice

1 Introduction

Un problème d'allocation de ressources multi-agents consiste à déterminer une distribution équitable et/ou efficace d'un ensemble de biens parmi un ensemble d'agents [9]. La plupart des travaux existants supposent que les biens sont alloués directement aux agents ou aux groupes qui les utiliseront, mais dans de nombreux contextes réels, les agents appartiennent à des organisations hiérarchiques : des groupes sont imbriqués dans des groupes plus vastes, formant une organisation hiérarchique multiniveaux. Une telle hiérarchie peut être naturellement modélisée sous la forme d'un arbre orienté, où chaque nœud est responsable de l'allocation du lot de biens qu'il reçoit à ses enfants.

Considérons l'allocation de bureaux à des chercheurs au sein d'une université présentant une structure hiérarchique typique : l'université est composée de départements, contenant chacun des laboratoires, qui comprennent à leur tour des groupes de recherche constitués de chercheurs individuels. Le processus d'allocation se déroule généralement

par étapes : les bureaux sont d'abord répartis entre les départements, puis entre les laboratoires au sein de chaque département, ensuite entre les groupes de recherche au sein de chaque laboratoire, et enfin entre les chercheurs individuels. Dans un tel cadre, l'équité doit être appliquée au sein de chaque niveau hiérarchique (par exemple, entre les départements ou entre les groupes de recherche d'un même laboratoire), plutôt qu'entre les différents niveaux, puisque les entités situées à des niveaux différents représentent des entités incomparables. Cet exemple s'étend à de nombreux problèmes d'allocation à travers des territoires et des organisations hiérarchiques. De plus, des applications peuvent émerger même en l'absence de hiérarchie explicitement définie. Par exemple, lors de l'attribution d'emplois à des personnes au chômage caractérisées par leur genre, leur âge et leur niveau d'études, on peut d'abord se concentrer sur l'équité de genre (par exemple, en accordant plus de poids aux femmes pour pallier leur sous-représentation), puis sur l'équité selon le niveau d'études pour les femmes (afin de promouvoir des femmes plus qualifiées) et sur l'âge pour les hommes en attribuant un poids égal à chaque tranche d'âge (pour équilibrer la représentation). De telles préférences hiérarchiques peuvent être naturellement représentées dans le cadre multiniveaux par un arbre orienté dont les nœuds sont pondérés. Notons que ce dernier exemple est étroitement lié aux politiques de discrimination positive du monde réel, telles que le système de quotas en Inde.

Différentes hypothèses peuvent être formulées concernant l'utilité des nœuds (agents) de l'arbre. Pour les nœuds internes, on pourrait supposer (i) que les agents ne se soucient que du lot qu'ils reçoivent et ignorent l'allocation ultérieure de ces biens; (ii) que les agents ont des préférences sur les propriétés de l'allocation à leurs enfants; ou (iii) que les agents ont des préférences qui dépendent des utilités effectives de leurs enfants. Dans cet article, nous adoptons l'hypothèse (iii), qui reflète l'idée que les entités internes n'agissent pas comme des agents mus par leur propre intérêt, mais plutôt comme des organes représentatifs de leurs membres. Par exemple, l'objectif d'un département est de maximiser la satisfaction globale des laboratoires qu'il contient, et de même, un laboratoire vise à maximiser la satisfaction de ses groupes de recherche.

Enfin, nous adoptons des utilités de rang de matroïde

*La version complète du papier est disponible ici : <https://arxiv.org/abs/2512.24105>

(matroid-rank) pour les feuilles, une classe de valuations bien établie qui capture de manière succincte les préférences pour la diversité des biens. Des exemples de telles préférences incluent des étudiants privilégiant des cursus couvrant plusieurs disciplines, ou des problèmes d'allocation avec des contraintes de diversité pouvant interdire le dépassement de quotas par catégorie [6]. Contrairement aux utilités additives, les valuations de rang de matroïde ont un pouvoir expressif limité sur les singletons, mais gagnent en richesse grâce à leur structure sous-modulaire, qui permet de capturer les complémentarités et les rendements décroissants. Une propriété remarquable des préférences de rang de matroïde est qu'elles permettent l'existence d'allocations satisfaisant simultanément l'efficacité et l'équité, comme l'ont observé plusieurs papiers [7, 5, 19]. Dans cet article, nous montrons que cette propriété souhaitable peut être étendue à notre cadre multiniveaux.

Allocation de vaccins. Une illustration concrète de notre modèle, combinant structure hiérarchique et préférences de rang de matroïde aux feuilles, est le problème de la distribution équitable des vaccins pendant la pandémie de COVID-19. Garantir l'équité territoriale importait non seulement pour des raisons morales, puisqu'en l'absence de coordination les pays riches peuvent s'assurer plus de doses que nécessaire au détriment des pays plus pauvres, mais aussi pour des raisons épidémiologiques, la vaccination à l'échelle mondiale étant essentielle pour empêcher une résurgence de l'épidémie. Des préoccupations similaires ont émergé à l'échelle nationale : en France, par exemple, certains parlementaires ont souligné des disparités dans l'allocation des vaccins entre les régions, certaines zones recevant significativement plus de doses que d'autres. De tels défis liés à la distribution territoriale équitable peuvent être naturellement traités dans le cadre que nous proposons, lequel permet d'intégrer des considérations d'équité à travers de multiples niveaux administratifs. En particulier, l'allocation devrait être équitable entre les pays, par exemple proportionnellement à leur population, puis récursivement équitable à travers les subdivisions territoriales successives au sein de chaque pays, pour descendre in fine au niveau des hôpitaux ou centres de vaccination.

De plus, les préférences des hôpitaux ou des centres de vaccination peuvent être naturellement modélisées à l'aide de fonctions de rang de matroïde. Considérons par exemple une situation dans laquelle un stock fixe de doses de vaccin doit être distribué entre des hôpitaux ou des centres de vaccination à l'échelle mondiale. Chaque hôpital a une capacité de stockage, limitant le nombre total de doses qu'il peut conserver en toute sécurité. En outre, les hôpitaux peuvent approuver ou désapprouver certains vaccins spécifiques ; pendant la pandémie de COVID-19, par exemple, certains pays ont refusé ou restreint l'utilisation de doses de vaccins provenant de certains fournisseurs étrangers. Par conséquent, chaque hôpital cherche à recevoir le plus grand nombre possible de doses parmi les vaccins qu'il approuve, tout en respectant ses contraintes de capacité.

Travaux connexes. Récemment, l'équité de groupe est devenue un sujet actif dans le domaine du partage équi-

table. Par exemple, [4, 14] ont étudié des notions d'équité de groupe exigeant que l'équité soit respectée entre n'importe quel groupe, et ont introduit différentes relaxations de l'absence d'envie de groupe (group envy-freeness). Lorsque les groupes sont prédéfinis, plusieurs notions et algorithmes ont également été proposés [6, 13, 15, 3]. Nous nous concentrons sur des feuilles ayant des valuations de rang de matroïde. Plusieurs articles étudient des problèmes classiques d'allocation (mono-niveau) où les agents possèdent de telles utilités [7, 5, 19]. En particulier, notre second algorithme étend le General Yankee Swap [19] à notre cadre multiniveaux. Les travaux de [16, 10, 2], qui cherchent à concilier les perspectives des groupes et des agents individuels, présentent également des similarités avec notre contribution. Plus précisément, [16] étudie l'absence d'envie individuelle et de groupe dans des configurations où les valuations des groupes dépendent des valuations des agents qui les composent, et proposent des algorithmes offrant des garanties à ces deux niveaux. En revanche, [10] font valoir que les agents situés à différents niveaux peuvent avoir des préférences distinctes, soulignant notamment que les préférences des groupes n'agrègent pas nécessairement les valuations de leurs membres (alors que nous supposons dans cet article que les nœuds agrègent l'utilité de leurs enfants). Enfin, [2] étudie un problème d'allocation bi-niveaux, motivé par le fonctionnement de certaines associations d'aide alimentaire. Contrairement au nôtre, leur article se concentre sur des mécanismes d'enchères. L'ensemble de ces travaux se limite toutefois à des environnements à deux niveaux. De plus, notre cadre présente des similitudes avec le récent modèle de répartition de sièges multiniveaux (multilevel apportionment) de [17]. En effet, leur modèle hiérarchique est similaire au nôtre, représentant la hiérarchie au moyen d'un arbre. Cependant, la portée de leur article se limite spécifiquement aux méthodes de répartition de sièges, un sous-domaine particulier de l'allocation de ressources. Notre modèle partage également des points communs avec les récents travaux sur l'équité locale [1, 8], dans la mesure où la structure hiérarchique induit une forme de contrainte de visibilité (par exemple, seuls les laboratoires rattachés au même département peuvent comparer leurs situations respectives).

Contributions. Dans cet article, nous introduisons formellement les problèmes d'allocation équitable multiniveaux et proposons deux algorithmes pour les résoudre. Nous supposons que les feuilles de l'arbre (par exemple, les groupes de recherche ou les personnes au chômage) possèdent des fonctions d'utilité de type rang de matroïde, et que chaque nœud interne vise à maximiser le bien-être utilitariste de ses enfants (c'est-à-dire que son utilité est égale à la somme des utilités de ses enfants). Le premier algorithme, introduit dans la Section 3, est une méthode descendante en temps polynomial satisfaisant plusieurs propriétés d'équité et d'efficacité. Cependant, comme discuté dans la Section 5, il peut s'avérer lent en pratique. Par conséquent, nous proposons un algorithme plus rapide dans la Section 4, étendant le General Yankee Swap [19], qui garantit l'efficacité et démontre une forte équité empirique.

2 Modèle

Nous considérons des problèmes d'allocation où un ensemble de m biens, noté $\mathcal{G} = \{g_1, \dots, g_m\}$, doit être distribué parmi des agents organisés selon une structure hiérarchique.

Structure hiérarchique. Nous considérons un problème d'allocation multiniveaux représenté par une arborescence (c'est-à-dire un arbre orienté enraciné dans lequel toutes les arêtes pointent à l'opposé de la racine), notée $\mathcal{T} = (\mathcal{N}, \mathcal{E})$. Ici, $\mathcal{N} = \{1, \dots, n\}$ est l'ensemble des nœuds représentant les agents, et \mathcal{E} est l'ensemble des arêtes orientées qui représentent les relations hiérarchiques entre eux. Nous supposons que les nœuds sont indexés selon un ordre topologique de \mathcal{T} , de sorte que le nœud 1 est la racine. De plus, chaque nœud $i \in \mathcal{N}$ est associé à un poids $w_i > 0$ représentant la prétention (*entitlement*) de l'agent i , tel que requis par certains critères d'équité. Soit $\mathcal{C}(i)$ l'ensemble des enfants du nœud i , défini par $\mathcal{C}(i) = \{j \in \mathcal{N} : (i, j) \in \mathcal{E}\}$. Soit $\mathcal{P}(i)$ le parent de i , c'est-à-dire l'unique nœud tel que $(\mathcal{P}(i), i) \in \mathcal{E}$. Nous notons $\text{Anc}(i)$ l'ensemble des ancêtres de i , c'est-à-dire les nœuds sur l'unique chemin allant de la racine 1 à i , en excluant i . Pour tout nœud i , soit $\mathcal{T}_i = (\mathcal{N}_i, \mathcal{E}_i)$ le sous-arbre de \mathcal{T} enraciné en i , constitué de tous les nœuds et arêtes appartenant aux chemins qui commencent en i . Soit $\mathcal{L}(i)$ l'ensemble des feuilles de \mathcal{T}_i , formellement défini par $\mathcal{L}(i) = \{j \in \mathcal{N}_i : \mathcal{C}(j) = \emptyset\}$. Soit $\mathcal{I}(i)$ l'ensemble des nœuds internes de \mathcal{T}_i , défini par $\mathcal{I}(i) = \mathcal{N}_i \setminus \mathcal{L}(i)$. Enfin, nous écrivons simplement $\mathcal{I} := \mathcal{I}(1)$ et $\mathcal{L} := \mathcal{L}(1)$ pour désigner, respectivement, les ensembles des nœuds internes et des feuilles de l'arbre tout entier.

Allocations. Nous introduisons maintenant les types d'allocation pertinents :

Définition 1 (allocation multiniveau). $\pi : \mathcal{N} \rightarrow 2^{\mathcal{G}}$ est une allocation multiniveaux si elle satisfait les propriétés suivantes : (i) $\pi(1) = \mathcal{G}$, (ii) $\pi(i) \supseteq \bigcup_{j \in \mathcal{C}(i)} \pi(j)$ pour tout $i \in \mathcal{I}$, et (iii) $\pi(i) \cap \pi(j) = \emptyset$ pour tous $i, j \in \mathcal{N}$ tel que $\mathcal{P}(i) = \mathcal{P}(j)$, où $\pi(i)$ dénote le lot d'un nœud $i \in \mathcal{N}$.

Nous exigeons que la racine possède tous les biens (c.-à-d. $\pi(1) = \mathcal{G}$) uniquement pour garantir qu'aucun n'est écarté a priori. Notons que nous n'imposons pas $\pi(i) = \bigcup_{j \in \mathcal{C}(i)} \pi(j)$ pour tout $i \in \mathcal{I}$, ce qui permet aux nœuds de ne pas allouer tous leurs biens, notamment pour satisfaire leur critère d'équité. À la place, nous imposons seulement $\pi(i) \supseteq \bigcup_{j \in \mathcal{C}(i)} \pi(j)$ pour rendre le modèle aussi général que possible. Par exemple, cela englobe les problèmes d'allocation avec charité [11], et cela s'aligne avec la plupart des études portant sur les valuations de rang de matroïde [7, 19], qui se restreignent aux allocations non redondantes (c'est-à-dire que seuls les biens ayant une utilité marginale positive sont attribués). Enfin, nous exigeons que chaque nœud interne alloue chacun de ses biens à au plus un de ses enfants. Soit Π l'ensemble de toutes les allocations multiniveaux.

Définition 2 (allocation multiniveaux restreinte). Pour une allocation multiniveaux $\pi \in \Pi$ et un ensemble de nœuds $N \subseteq \mathcal{N}$, la restriction de π à N est $\pi|_N = (\pi(i))_{i \in N}$.

Définition 3 (allocation locale). Étant donné un ensemble de nœuds $N \subseteq \mathcal{N}$ et un ensemble de biens $S \subseteq \mathcal{G}$, $A : N \rightarrow 2^S$ est une allocation locale si $\bigcup_{i \in N} A(i) \subseteq S$ et $A(i) \cap A(j) = \emptyset$ pour tous $i, j \in N$.

Pour tout $N \subseteq \mathcal{N}$ et $S \subseteq \mathcal{G}$, soit \mathcal{A}_N^S l'ensemble des allocations locales correspondantes. Notons que pour toute allocation multiniveaux $\pi \in \Pi$ et tout $i \in \mathcal{I}$, l'allocation restreinte $\pi|_{\mathcal{C}(i)}$ est une allocation locale appartenant à $\mathcal{A}_{\mathcal{C}(i)}^{\pi(i)}$.

Modèle d'utilité. Soit $v_i : \Pi \rightarrow \mathbb{R}_{\geq 0}$ la fonction d'utilité du nœud $i \in \mathcal{N}$, et soit $v = (v_i)_{i \in \mathcal{N}}$. Notons que la fonction d'utilité est définie sur l'ensemble des allocations multiniveaux (plutôt que sur des lots de biens), puisqu'une utilité d'un nœud peut dépendre de la manière dont les biens sont alloués à ses enfants, ou à tout autre nœud de l'arbre. Dans cet article, pour tout nœud interne $i \in \mathcal{I}$ et toute allocation multiniveaux $\pi \in \Pi$, $v_i(\pi)$ quantifie un certain concept de bien-être global dérivé des enfants de i à partir de π . Nous nous concentrons ici sur le bien-être utilitariste, c'est-à-dire $v_i(\pi) := \sum_{j \in \mathcal{C}(i)} v_j(\pi)$. Par conséquent, nous supposons que les nœuds internes ont des utilités additives sur leurs enfants. Cette hypothèse est réaliste dans de nombreuses applications où les nœuds internes représentent des entités dont l'utilité découle uniquement de la satisfaction de leurs membres. Remarquons que, par linéarité de la somme, $v_i(\pi)$ peut être réécrit $v_i(\pi) = \sum_{x \in \mathcal{L}(i)} v_x(\pi)$ où la somme porte sur les feuilles du sous-arbre \mathcal{T}_i . Dans certaines applications, nous pourrions envisager des modèles plus complexes intégrant une composante personnelle en plus de la dépendance envers les enfants. Nous laissons cette extension pour des travaux futurs.

En revanche, l'utilité de toute feuille $x \in \mathcal{L}$ ne dépend que du lot de biens qu'elle reçoit (puisque elle n'a pas d'enfants). Par conséquent, il existe une fonction $u_x : 2^{\mathcal{G}} \rightarrow \mathbb{R}_{\geq 0}$ telle que $v_x(\pi) = u_x(\pi(x))$ pour toute allocation multiniveaux $\pi \in \Pi$, et nous supposons que $u_x(\emptyset) = 0$. Nous supposons également que u_x appartient à la classe des fonctions de rang de matroïde, également connues sous le nom d'utilités sous-modulaires binaires.

Définition 4 (fonction de rang de matroïde). Une fonction d'ensemble $u : 2^{\mathcal{G}} \rightarrow \mathbb{R}_{\geq 0}$ est une Fonction de Rang de Matroïde (MRF) si elle satisfait : (i) Monotonie : $u(S) \leq u(T)$ pour tout $S \subseteq T \subseteq \mathcal{G}$, (ii) Sous-modularité : $\Delta^u(S, g) \geq \Delta^u(T, g)$ pour tout $S \subseteq T \subseteq \mathcal{G}$ et tout $g \in \mathcal{G} \setminus T$, et (iii) Gain marginal binaire : $\Delta^u(S, g) \in \{0, 1\}$ pour tout $S \subseteq \mathcal{G}$ et tout $g \in \mathcal{G} \setminus S$, où $\Delta^u(S, g) = u(S \cup \{g\}) - u(S)$ est le gain marginal de g étant donné S , pour tout $S \subseteq \mathcal{G}$ et tout $g \in \mathcal{G} \setminus S$.

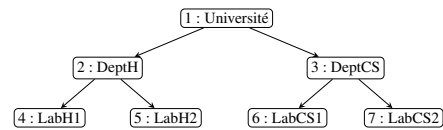


FIGURE 1 – La structure hiérarchique d'une université.

Exemple 1. Une université doit allouer 6 bureaux $\mathcal{G} = \{a, b, c, d, e, f\}$ à deux départements (Humanités et Informatique), qui à leur tour les distribueront parmi leurs laboratoires respectifs (voir Fig. 1).

Les laboratoires sont les feuilles. Leurs utilités type rang de matroïde sont les suivantes : LabH1 obtient une utilité de 1 s'il reçoit n'importe quel bureau; LabH2 se comporte de manière similaire, à l'exception du bureau a qu'il ne désire pas; LabCS1 et LabCS2 ont des valuations additives binaires et approuvent tous les biens. Supposons que DeptH et DeptCS reçoivent $\{a, b\}$ et $\{c, d, e, f\}$ respectivement, puis attribuent leurs biens de sorte que LabH1, LabH2, et LabCS1 reçoivent $\{a\}$, $\{b\}$, et $\{c, d, e, f\}$, respectivement, laissant LabCS2 avec un lot vide. Cette allocation multiniveaux, notée π , vérifie :

Nœud i	1	2	3	4	5	6	7
$\pi(i)$	$\{a, b, c, d, e, f\}$	$\{a, b\}$	$\{c, d, e, f\}$	$\{a\}$	$\{b\}$	$\{c, d, e, f\}$	\emptyset
$v_i(\pi)$	6	2	4	1	1	4	0

On pourrait se demander si l'allocation effectuée par l'université est efficace et équitable, dans un certain sens, vis-à-vis des départements, en considérant l'allocation restreinte $\pi|_{\mathcal{C}(1)}$ (qui est l'allocation locale dans $\mathcal{A}_{\mathcal{C}(1)}^G$) définie par $\pi|_{\mathcal{C}(1)}(2) = \{a, b\}$ et $\pi|_{\mathcal{C}(1)}(3) = \{c, d, e, f\}$.

Notre objectif est de calculer une allocation multiniveaux qui garantisse simultanément un bien-être élevé et satisfasse un critère d'équité vis-à-vis des enfants de chaque nœud (interne).

Critères d'efficacité. Nous introduisons maintenant les notions d'efficacité considérées dans cet article.

Définition 5 (non-redondance par rapport à v). *L'allocation multiniveaux $\pi \in \Pi$ est non redondante si $v_i(\pi) - v_i(\pi_{-g}) > 0$ pour tout $i \in \mathcal{N}$ et tout $g \in \pi(i)$, où π_{-g} désigne l'allocation multiniveaux obtenue à partir de π en retirant g de tous les lots $\pi(i)$, $i \in \mathcal{N}$, qui incluent g .*

Définition 6 (optimalité utilitariste multiniveaux par rapport à v). *Étant donné une allocation multiniveaux quelconque $\pi \in \Pi$ et un nœud interne quelconque $i \in \mathcal{I}$, $\pi|_{\mathcal{C}(i)}$ est optimale au sens utilitariste par rapport à v si et seulement si pour toute $\pi' \in \Pi$ telle que $\pi'(i) = \pi(i)$, nous avons :*

$$\sum_{j \in \mathcal{C}(i)} v_j(\pi) \geq \sum_{j \in \mathcal{C}(i)} v_j(\pi')$$

$\pi \in \Pi$ est dite optimale au sens utilitariste multiniveaux par rapport à v si $\pi|_{\mathcal{C}(i)}$ est optimale au sens utilitariste par rapport à v pour tout $i \in \mathcal{I}$.

Notons qu'une allocation optimale au sens utilitariste multiniveaux peut être calculée en temps polynomial, grâce à l'additivité de v_i , pour tout $i \in \mathcal{I}$. En effet, il suffit de trouver une allocation locale $A \in \mathcal{A}_{\mathcal{C}(i)}^G$ qui maximise $\sum_{x \in \mathcal{L}} u_x(A(x))$, puis de définir $\pi(x) = A(x)$ pour toutes les feuilles $x \in \mathcal{L}$ et $\pi(i) = \cup_{x \in \mathcal{C}(i)} A(x)$ pour chaque nœud interne $i \in \mathcal{I}$. Puisque chaque u_x est une fonction de rang de matroïde, A peut être calculée en temps polynomial : ce problème a déjà été abordé en partage équitable par des articles antérieurs [7, 5], qui montrent qu'il peut être résolu en temps polynomial en le réduisant à un problème d'intersection de matroïdes [12]. Dans la preuve du Théorème 2, nous montrons qu'il peut être résolu encore plus rapidement en utilisant le problème de partition de matroïdes, dont le meilleur algorithme disponible a une

complexité inférieure à celle du problème d'intersection de matroïdes [18]. Néanmoins, nous soulignons qu'une allocation optimale du point de vue utilitariste multiniveaux n'est pas nécessairement équitable parmi les enfants de chaque nœud (voir Exemple 2).

Critères d'équité. Nous considérons des critères d'équité comparant les allocations à travers leurs vecteurs d'utilité. Dans cet article, nous considérons un ensemble \mathcal{F} de quatre critères d'équité bien connus qui permettent la comparaison de n'importe quelle paire de vecteurs d'utilité, induisant ainsi un ordre total \succeq sur les allocations locales possibles : la dominance de Lorenz, le bien-être de Nash pondéré maximum, le leximin pondéré, et le bien-être des moyennes pondérées d'ordre p maximum. Nous définissons un critère d'équité spécifique au nœud Ψ_i pour chaque nœud interne $i \in \mathcal{I}$. Dans notre cadre, cela revient à comparer les vecteurs d'utilité des allocations locales au nœud i . Spécifiquement, étant donné une allocation multiniveaux $\pi \in \Pi$, le vecteur d'utilité de $\pi|_{\mathcal{C}(i)}$ est défini par $\overrightarrow{v_i(\pi|_{\mathcal{C}(i)})} = (v_{i_1}(\pi), v_{i_2}(\pi), \dots, v_{i_p}(\pi))$ où i_1, \dots, i_p sont les enfants de i , ordonnés par leur indice. Étant donné deux allocations multiniveaux $\pi, \pi' \in \Pi$, nous disons que $\pi|_{\mathcal{C}(i)}$ est plus équitable par rapport à Ψ_i et v que $\pi'|_{\mathcal{C}(i)}$ si

$$\overrightarrow{v_i(\pi|_{\mathcal{C}(i)})} \succeq_{\Psi_i} \overrightarrow{v_i(\pi'|_{\mathcal{C}(i)})}$$

également noté par $\pi|_{\mathcal{C}(i)} \succeq_{\Psi_i}^v \pi'|_{\mathcal{C}(i)}$ par concision. Soit $\Psi = (\Psi_i)_{i \in \mathcal{I}}$. Nous définissons maintenant la notion d'équité multiniveaux associée.

Définition 7 (Ψ -maximisante multiniveaux par rapport à v). *Étant donné une allocation multiniveaux quelconque $\pi \in \Pi$ et un nœud interne quelconque $i \in \mathcal{I}$, $\pi|_{\mathcal{C}(i)}$ est dite Ψ_i -maximisante par rapport à v si pour toute $\pi' \in \Pi$ telle que $\pi'(i) = \pi(i)$, nous avons :*

$$\pi|_{\mathcal{C}(i)} \succeq_{\Psi_i}^v \pi'|_{\mathcal{C}(i)}$$

$\pi \in \Pi$ est Ψ -maximisante multiniveaux par rapport à v si $\pi|_{\mathcal{C}(i)}$ est Ψ_i -maximisante par rapport à v pour tout $i \in \mathcal{I}$.

Nous définissons ensuite les quatre notions d'équité considérées dans l'article, adaptées à notre cadre multiniveaux.

Définition 8 (dominance de Lorenz par rapport à v). *Étant donné une allocation multiniveaux $\pi \in \Pi$ et un nœud interne $i \in \mathcal{I}$ dont les enfants sont $\{i_1, \dots, i_p\}$, $\pi|_{\mathcal{C}(i)}$ est dominante au sens de Lorenz par rapport à v si pour toute $\pi' \in \Pi$ telle que $\pi'(i) = \pi(i)$, nous avons :*

$$\forall k \in \{1, \dots, p\}, \sum_{t=1}^k \overrightarrow{s(\pi|_{\mathcal{C}(i)})}_t \geq \sum_{t=1}^k \overrightarrow{s(\pi'|_{\mathcal{C}(i)})}_t$$

où $\overrightarrow{s(\cdot)}$ est le vecteur $\overrightarrow{v_i(\cdot)}$ trié par ordre croissant, et $\overrightarrow{s(\cdot)}_t$ est la $t^{\text{ème}}$ composante de $\overrightarrow{s(\cdot)}$ pour tout $t \in \{1, \dots, p\}$.

Définition 9 (bien-être de Nash pondéré maximum par rapport à v). *Étant donné une allocation multiniveaux $\pi \in \Pi$ et un nœud interne $i \in \mathcal{I}$, $\pi|_{\mathcal{C}(i)}$ maximise le bien-être de Nash pondéré par rapport à v si elle minimise le nombre d'enfants ayant une utilité nulle, et sous cette contrainte,*

satisfait pour toute $\pi' \in \Pi$ telle que $\pi'(i) = \pi(i)$:

$$\prod_{j \in \mathcal{C}(i)} v_j(\pi)^{w_j} \geq \prod_{j \in \mathcal{C}(i)} v_j(\pi')^{w_j}$$

Définition 10 (leximin pondéré par rapport à v). *Étant donné une allocation multiniveaux $\pi \in \Pi$ et un nœud interne $i \in \mathcal{I}$ dont les enfants sont $\{i_1, \dots, i_p\}$, $\pi|_{\mathcal{C}(i)}$ est leximin pondérée par rapport à v s'il n'existe aucune $\pi' \in \Pi$ avec $\pi'(i) = \pi(i)$ telle que :*

$$\overrightarrow{e(\pi'|_{\mathcal{C}(i)})} \succ_{lex} \overrightarrow{e(\pi|_{\mathcal{C}(i)})}$$

où $\overrightarrow{e(\cdot)}$ est le vecteur $\left(\frac{v_i(\cdot)_1}{w_{i_1}}, \dots, \frac{v_i(\cdot)_p}{w_{i_p}}\right)$ trié par ordre croissant, et \succ_{lex} est la dominance lexicographique¹.

Définition 11 (bien-être des moyennes pondérées d'ordre p maximum par rapport à v). *Étant donné une allocation multiniveaux $\pi \in \Pi$ et un nœud interne $i \in \mathcal{I}$, $\pi|_{\mathcal{C}(i)}$ maximise le bien-être des moyennes pondérées d'ordre p par rapport à v pour un certain $p \leq 1$ si elle minimise le nombre d'enfants ayant une utilité nulle, et sous cette contrainte satisfait pour toute $\pi' \in \Pi$ telle que $\pi'(i) = \pi(i)$:*

$$\left(\sum_{j \in \mathcal{C}(i)} w_j v_j(\pi)^p\right)^{\frac{1}{p}} \geq \left(\sum_{j \in \mathcal{C}(i)} w_j v_j(\pi')^p\right)^{\frac{1}{p}}$$

Ces quatre notions sont compatibles avec l'optimisation du bien-être utilitariste lorsque les fonctions de valuation sont des MRF [7, 19].

Nous illustrons maintenant le manque potentiel d'équité parmi les enfants d'un nœud dans le cadre d'allocations optimales du point de vue utilitariste multiniveaux, soulignant la nécessité de concevoir des algorithmes qui tiennent compte à la fois de l'optimisation du bien-être et de l'équité.

Exemple 2. *Nous utilisons la même instance que l'Exemple 1. Comparons les allocations multiniveaux suivantes (nous ne reportons que les lots des laboratoires, car les lots des départements peuvent ensuite être obtenus en prenant l'union des lots de leurs laboratoires respectifs) :*

Alloc	LabH1	LabH2	LabCS1	LabCS2	utilities
π	{a}	{b}	{c, d, e, f}	\emptyset	(1, 1, 4, 0)
π'	{b}	{a}	{c, d, e, f}	\emptyset	(1, 0, 4, 0)
π''	{a}	{b}	{c, d}	{e, f}	(1, 1, 2, 2)

Tout d'abord, remarquons que π' n'est pas optimale au sens utilitariste pour DeptH, car échanger a et b entre LabH1 et LabH2 donnerait une utilité de 2 au lieu de 1. Ensuite, on peut montrer que π est optimale au sens utilitariste multiniveaux mais n'est pas équitable au nœud DeptCS en considérant la dominance de Lorenz par rapport à v car elle est dominée par π'' . Enfin, π'' est à la fois optimale au sens utilitariste multiniveaux et dominante au sens de Lorenz multiniveaux par rapport à v .

3 Algorithme Séquentiel Multiniveaux

Nous proposons maintenant un algorithme séquentiel multiniveaux qui suit une approche descendante (*top-down*),

1. Given two vectors $x, y \in \mathbb{R}^c$ for some positive integer c , x lexicographically dominates y if there exists $k \in \{1, \dots, c\}$ such that $x_j = y_j$ for all $j \in \{1, \dots, k-1\}$ and $x_k > y_k$.

s'inspirant des systèmes hiérarchiques de la vie réelle (par exemple, dans les systèmes de déploiement de vaccins, les doses sont achetées de manière centralisée puis distribuées progressivement le long de la hiérarchie administrative, de l'État aux régions, des régions aux départements, et ainsi de suite). En commençant par la racine, à laquelle est attribué l'ensemble \mathcal{G} , nous calculons une allocation locale aux enfants de la racine. Ensuite, pour chacun de ses enfants, nous calculons une allocation locale du lot qu'il a reçu de la racine vers ses propres enfants, et ainsi de suite jusqu'aux feuilles. Nous montrerons qu'une telle approche permet de construire une allocation Ψ -maximisante multiniveaux et optimale au sens utilitariste multiniveaux par rapport à v .

Utilité estimée. Afin d'atteindre l'efficacité et l'équité à chaque nœud, chacun doit prendre en compte les utilités de ses enfants lorsqu'il leur attribue des lots de biens. Cependant, dans notre cadre, chaque nœud interne $i \in \mathcal{I}$ possède une fonction d'utilité v_i qui est définie sur l'ensemble des allocations multiniveaux possibles, et non sur l'ensemble de tous les lots. En effet, l'utilité de chaque enfant dépend non seulement du lot qu'il reçoit, mais aussi de la manière dont ce lot sera alloué à ses propres enfants (ce qui dépend également des notions d'équité considérées). Par conséquent, dans le cadre d'une procédure descendante, l'utilité d'un nœud interne est indéterminée jusqu'à ce que les feuilles soient atteintes, ce qui souligne une difficulté clé inhérente à l'approche descendante. Pour remédier à ce problème, notre algorithme associe à chaque nœud $i \in \mathcal{N}$ une fonction d'utilité $\hat{v}_i : \mathcal{2}^{\mathcal{G}} \rightarrow \mathbb{R}_{\geq 0}$ estimant l'utilité $v_i(\pi^*)$ obtenue par i à partir de l'allocation multiniveaux $\pi^* \in \Pi$ finalement renvoyée par l'algorithme. Cette fonction d'utilité estimée est utilisée comme un substitut de v_i . Pour tout $S \subseteq \mathcal{G}$, la valeur $\hat{v}_i(S)$ est définie en utilisant les feuilles de \mathcal{T}_i comme suit :

$$\hat{v}_i(S) := \begin{cases} \max_{A \in \mathcal{A}_{\mathcal{L}(i)}^S} \sum_{x \in \mathcal{L}(i)} u_x(A(x)), & \text{si } i \in \mathcal{I}, \\ u_i(S), & \text{sinon.} \end{cases}$$

Nous montrons dans la preuve du Théorème 2 que $\hat{v}_i(S)$ peut être calculée efficacement. Pour garantir que π^* est optimale au sens utilitariste multiniveaux et Ψ -maximisante multiniveaux par rapport à v , notre algorithme construit une allocation multiniveaux qui est optimale au sens utilitariste multiniveaux et Ψ -maximisante multiniveaux par rapport à \hat{v} en identifiant séquentiellement, à chaque $i \in \mathcal{I}$, une allocation locale qui est optimale au sens utilitariste et Ψ_i -maximisante par rapport à \hat{v} . Nous donnons les définitions de ces concepts basés sur \hat{v} ci-dessous.

Définition 12 (optimalité utilitariste multiniveaux par rapport à \hat{v}). *Étant donné un lot de biens $S \subseteq \mathcal{G}$ et un nœud interne quelconque $i \in \mathcal{I}$, l'allocation locale $A \in \mathcal{A}_{\mathcal{C}(i)}^S$ est optimale au sens utilitariste par rapport à \hat{v} si, pour toute $A' \in \mathcal{A}_{\mathcal{C}(i)}^S$, nous avons :*

$$\sum_{j \in \mathcal{C}(i)} \hat{v}_j(A(j)) \geq \sum_{j \in \mathcal{C}(i)} \hat{v}_j(A'(j))$$

$\pi \in \Pi$ est optimale au sens utilitariste multiniveaux par rapport à \hat{v} si, pour tout $i \in \mathcal{I}$, $\pi|_{\mathcal{C}(i)}$ est optimale au sens utilitariste par rapport à \hat{v} pour $S = \pi(i)$.

Étant donné un ensemble de biens $S \subseteq \mathcal{G}$ et un ensemble de nœuds $N \subseteq \mathcal{N}$, nous définissons le vecteur d'utilité estimée d'une allocation locale $A \in \mathcal{A}_N^S$ comme $\hat{v}_i(A) = (\hat{v}_{i_1}(A(i_1)), \hat{v}_{i_2}(A(i_2)), \dots, \hat{v}_{i_p}(A(i_p)))$ où i_1, \dots, i_p sont les nœuds de N ordonnés par leur indice. À chaque nœud interne $i \in \mathcal{I}$ avec le lot $S \subseteq \mathcal{G}$, nous utilisons la notation $\succeq_{\Psi_i}^{\hat{v}}$ pour comparer deux allocations locales $A, B \in \mathcal{A}_{\mathcal{C}(i)}^S$ en nous basant sur les vecteurs d'utilité $\hat{v}_i(A)$ et $\hat{v}_i(B)$. Cela conduit aux notions d'équité multiniveaux suivantes.

Définition 13 (Ψ -maximisation multiniveaux par rapport à \hat{v}). Étant donné un lot de biens $S \subseteq \mathcal{G}$ et un nœud interne $i \in \mathcal{I}$, l'allocation locale $A \in \mathcal{A}_{\mathcal{C}(i)}^S$ est Ψ_i -maximisante par rapport à \hat{v} si, pour toute $A' \in \mathcal{A}_{\mathcal{C}(i)}^S$, nous avons :

$$A \succeq_{\Psi_i}^{\hat{v}} A'$$

$\pi \in \Pi$ est dite Ψ -maximisante multiniveaux par rapport à \hat{v} si, pour tout $i \in \mathcal{I}$, $\pi|_{\mathcal{C}(i)}$ est Ψ_i -maximisante par rapport à \hat{v} pour $S = \pi(i)$.

Il convient de noter la distinction entre les Déf. 6, 7 et les Déf. 12, 13 : ces dernières sont formulées en termes de \hat{v} plutôt que de v , et sont définies par rapport aux allocations locales $A \in \mathcal{A}_{\mathcal{C}(i)}^S$ par opposition aux allocations multiniveaux. Les définitions des notions d'équité dans \mathcal{F} définies par rapport à \hat{v} sont dans la version complète du papier.

Algorithme. Nous pouvons maintenant formaliser le fonctionnement de l'algorithme. En commençant au nœud $i = 1$, auquel est attribué l'ensemble \mathcal{G} , nous calculons une allocation locale $A^i \in \mathcal{A}_{\mathcal{C}(i)}^{\mathcal{G}}$ qui est optimale au sens utilitariste et Ψ_i -maximisante par rapport à \hat{v} . Chaque enfant de la racine $j \in \mathcal{C}(i)$ reçoit ainsi le lot de biens $A^i(j)$ et calculera à son tour une allocation locale dans $\mathcal{A}_{\mathcal{C}(j)}^{A^i(j)}$ pour ses propres enfants, qui est optimale au sens utilitariste et Ψ_j -maximisante par rapport à \hat{v} . Le processus se répète jusqu'à atteindre les feuilles. Cet Algorithme Séquentiel Multiniveaux (**SMA**) construit une allocation multiniveaux à partir de l'appel initial **SMA**(1, π , v , Ψ) où π est telle que $\pi(1) = \mathcal{G}$ et $\pi(i) = \emptyset$ pour tout $i \in \mathcal{N} \setminus \{1\}$. Le pseudo-code est dans la version complète du papier. Nous illustrons maintenant une exécution simple de **SMA**.

Exemple 3. Considérons une instance avec $m = 5$ biens, c.-à-d. $\mathcal{G} = \{g_1, \dots, g_5\}$, et $n = 7$ agents disposés en un arbre binaire équilibré : $\mathcal{C}(1) = \{2, 3\}$, $\mathcal{C}(2) = \{4, 5\}$, et $\mathcal{C}(3) = \{6, 7\}$ (comme dans la Fig. 1). Les agents ont des poids $w_2 = 5$, $w_3 = 2$, et $w_i = 1$ pour $i \in \{4, 5, 6, 7\}$. Soit Ψ_1 le bien-être de Nash pondéré maximum et Ψ_2, Ψ_3 la dominance de Lorenz. Les utilités des feuilles sont additives binaires : les agents 4, 5 et 6 n'accordent de la valeur qu'aux biens g_1, g_2, g_3 tandis que l'agent 7 n'accorde de la valeur qu'à g_3, g_4, g_5 . L'algorithme commence par l'appel **SMA**(1, π , v , Ψ), avec $\pi(1) = \mathcal{G}$ et $\pi(i) = \emptyset$ pour tout $i \neq 1$. Il calcule une allocation locale optimale au sens utilitariste $A^1 \in \mathcal{A}_{\mathcal{C}(1)}^{\mathcal{G}}$ maximisant le bien-être de Nash pondéré par rapport à \hat{v} , par ex., $A^1(2) = \{g_1, g_2, g_3\}$ et $A^1(3) = \{g_4, g_5\}$. Nous mettons à jour π en $\pi(2) = \{g_1, g_2, g_3\}$ et $\pi(3) = \{g_4, g_5\}$. **SMA**(2, π , v , Ψ) calcule

une allocation locale $A^2 \in \mathcal{A}_{\mathcal{C}(2)}^{\pi(2)}$ qui est optimale au sens utilitariste et dominante de Lorenz par rapport à \hat{v} , par ex., $A^2(4) = \{g_1, g_3\}$ et $A^2(5) = \{g_2\}$, et nous fixons $\pi(4) = \{g_1, g_3\}$ et $\pi(5) = \{g_2\}$. **SMA**(3, π , v , Ψ) calcule une allocation locale $A^3 \in \mathcal{A}_{\mathcal{C}(3)}^{\pi(3)}$ avec les mêmes propriétés. Puisque le nœud 6 ne tire aucune valeur de g_4 et g_5 , nous obtenons $A^3(6) = \emptyset$ et $A^3(7) = \{g_4, g_5\}$. Nous mettons à jour π en conséquence. Puisque les feuilles ne récurseront pas, l'allocation finale est : $\pi(1) = \mathcal{G}$, $\pi(2) = \{g_1, g_2, g_3\}$, $\pi(3) = \{g_4, g_5\}$, $\pi(4) = \{g_1, g_3\}$, $\pi(5) = \{g_2\}$, $\pi(6) = \emptyset$ et $\pi(7) = \{g_4, g_5\}$.

Pour prouver que **SMA** peut être implémenté en temps polynomial, nous montrons d'abord que les utilités estimées \hat{v}_i , pour $i \in \mathcal{N}$, sont des MRFs. Les preuves manquantes sont disponibles dans la version complète du papier.

Proposition 1. \hat{v}_i est une MRF pour chaque nœud $i \in \mathcal{N}$.

Proof sketch. Nous montrons que, pour tout $i \in \mathcal{I}$, la fonction \hat{v}_i coïncide avec la fonction de rang du matroïde union formé à partir des matroïdes associés aux feuilles $\mathcal{L}(i)$. \square

Nous montrons ensuite que **SMA** a une complexité temporelle polynomiale. Dans cette preuve (disponible dans la version complète du papier), nous montrons que le calcul de $\hat{v}_i(\cdot)$ pour tout $i \in \mathcal{I}$ se réduit à un problème de partition de matroïdes, qui peut être résolu en temps polynomial et présente une complexité inférieure à l'approche standard d'intersection de matroïdes pour calculer une allocation optimale non redondante.

Théorème 2. L'algorithme **SMA** peut être implémenté en temps polynomial par rapport à n et m , lorsque $\Psi_i \in \mathcal{F}$ pour tout $i \in \mathcal{I}$.

Soit π^* l'allocation multiniveaux renvoyée par **SMA**. Pour montrer que π^* est optimale au sens utilitariste multiniveaux et Ψ -maximisante multiniveaux par rapport à v , notons d'abord que cela est vrai par rapport à \hat{v} .

Lemme 3. π^* est à la fois optimale au sens utilitariste multiniveaux et Ψ -maximisante multiniveaux par rapport à \hat{v} .

Démonstration. À la Ligne 5, nous avons $\pi^*|_{\mathcal{C}(i)} = A^i$ pour tous les nœuds $i \in \mathcal{I}$. Puisque A^i est à la fois optimale au sens utilitariste et Ψ_i -maximisante par rapport à \hat{v} pour $S = \pi(i)$, π^* est optimale au sens utilitariste multiniveaux et Ψ -maximisante multiniveaux par rapport à \hat{v} (par les Déf. 12 et 13). \square

La proposition suivante relie \hat{v} et v afin d'étendre ce résultat aux valuations d'origine.

Proposition 4. Pour tout $i \in \mathcal{N}$, $\hat{v}_i(\pi^*(i)) = v_i(\pi^*)$, et $\hat{v}_i(\pi(i)) \geq v_i(\pi)$ pour toute allocation $\pi \in \Pi$.

Théorème 5. **SMA** renvoie une allocation multiniveaux qui est à la fois optimale au sens utilitariste multiniveaux et Ψ -maximisante multiniveaux par rapport à v .

Démonstration. Par les Déf. 6 et 7, il nous suffit de prouver que $\pi^*|_{\mathcal{C}(i)}$ est optimale au sens utilitariste et Ψ_i -maximisante par rapport à v pour tout $i \in \mathcal{I}$. Tout d'abord, rappelons que $\pi^*|_{\mathcal{C}(i)}$ est optimale au sens utilitariste par rapport à \hat{v} pour $S = \pi(i)$ (voir Lemme 3). Par conséquent, $\sum_{j \in \mathcal{C}(i)} \hat{v}_j(\pi^*(j)) \geq \sum_{j \in \mathcal{C}(i)} \hat{v}_j(A(j))$ est vraie pour toute $A \in \mathcal{A}_{\mathcal{C}(i)}^{\pi^*(i)}$. Ensuite, en utilisant la première partie de la Prop. 4, nous obtenons $\sum_{j \in \mathcal{C}(i)} v_j(\pi^*) \geq \sum_{j \in \mathcal{C}(i)} \hat{v}_j(A(j))$, ce qui implique trivialement $\sum_{j \in \mathcal{C}(i)} v_j(\pi^*) \geq \sum_{j \in \mathcal{C}(i)} \hat{v}_j(\pi(j))$ pour toute $\pi \in \Pi$ telle que $\pi|_{\mathcal{C}(i)} \in \mathcal{A}_{\mathcal{C}(i)}^{\pi^*(i)}$. Nous pouvons alors déduire $\sum_{j \in \mathcal{C}(i)} v_j(\pi^*) \geq \sum_{j \in \mathcal{C}(i)} \hat{v}_j(\pi(j)) \geq \sum_{j \in \mathcal{C}(i)} v_j(\pi(j))$ à partir de la seconde partie de la Prop. 4, montrant que $\pi^*|_{\mathcal{C}(i)}$ est optimale au sens utilitariste par rapport à v . Nous montrons maintenant que $\pi^*|_{\mathcal{C}(i)}$ est Ψ_i -maximisante par rapport à v . Puisque $\pi^*|_{\mathcal{C}(i)}$ est Ψ_i -maximisante par rapport à \hat{v} pour $S = \pi(i)$ (par le Lemme 3), $\overrightarrow{\hat{v}_i(\pi^*|_{\mathcal{C}(i)})} \succeq_{\Psi_i} \overrightarrow{\hat{v}_i(A)}$ est vraie pour toute $A \in \mathcal{A}_{\mathcal{C}(i)}^{\pi^*(i)}$. Par la Prop. 4, $\overrightarrow{\hat{v}_j(\pi^*(j))} = \overrightarrow{v_j(\pi^*)}$ est vraie pour tout $j \in \mathcal{C}(i)$, et donc $\overrightarrow{\hat{v}_i(\pi^*|_{\mathcal{C}(i)})} = \overrightarrow{v_i(\pi^*|_{\mathcal{C}(i)})}$. Ainsi pour toute $A \in \mathcal{A}_{\mathcal{C}(i)}^{\pi^*(i)}$, nous avons $\overrightarrow{v_i(\pi^*|_{\mathcal{C}(i)})} \succeq_{\Psi_i} \overrightarrow{v_i(A)}$, ce qui implique $\overrightarrow{v_i(\pi^*|_{\mathcal{C}(i)})} \succeq_{\Psi_i} \overrightarrow{\hat{v}_i(\pi|_{\mathcal{C}(i)})}$ pour toute $\pi \in \Pi$ telle que $\pi|_{\mathcal{C}(i)} \in \mathcal{A}_{\mathcal{C}(i)}^{\pi^*(i)}$. Ensuite, en utilisant la seconde partie de la Prop. 4, nous obtenons $\overrightarrow{\hat{v}_i(\pi|_{\mathcal{C}(i)})} \succeq_{\Psi_i} \overrightarrow{v_i(\pi|_{\mathcal{C}(i)})}$. Nous obtenons alors $\overrightarrow{v_i(\pi^*|_{\mathcal{C}(i)})} \succeq_{\Psi_i} \overrightarrow{v_i(\pi|_{\mathcal{C}(i)})}$. \square

SMA est général au sens où il peut être implémenté en utilisant n'importe quel algorithme conçu pour le cadre standard (mono-niveau) afin de calculer A^i au nœud $i \in \mathcal{I}$, et il fonctionne avec d'autres critères d'équité, à condition qu'il existe une allocation optimale au sens utilitariste qui soit également équitable. Par exemple, nous montrons dans la version complète du papier que **SMA** peut calculer une allocation multiniveaux sans envie à un bien près, en utilisant l'algorithme de [7] et une adaptation appropriée de l'absence d'envie à notre cadre multiniveaux. Cependant, bien que **SMA** puisse s'exécuter en temps polynomial pour divers critères d'équité, il reste coûteux dans la pratique, comme nous le montrons dans la Section 5.

4 General Yankee Swap Multiniveaux

Dans cette section, nous proposons une extension multiniveaux du General Yankee Swap (**GYS**) [19], noté **MGYS**. **GYS** s'applique aux notions d'équité satisfaisant deux propriétés : (a) les allocations équitables sont Pareto-optimales ; (b) le critère d'équité possède une *fonction de gain* cohérente, décroissante par rapport à l'utilité d'un agent, qui indique quel agent doit être priorisé. Les notions répondant à ces critères incluent la dominance de Lorenz, le leximin pondéré, le bien-être de Nash pondéré maximum et le bien-être des moyennes pondérées d'ordre p maximum. Bien que **MGYS** produise toujours une allocation optimale au sens utilitariste multiniveaux par rapport à v , elle n'est pas nécessairement Ψ -maximisante multiniveaux et sert donc d'heuristique. Néanmoins, comme le montre

la Section 5, il est beaucoup plus rapide que **SMA** tout en atteignant une forte équité en pratique.

Fondamentalement, **MGYS** revient à appliquer **GYS** aux feuilles de l'arbre, avec deux modifications : (i) nous proposons un nouveau critère adapté au cadre multiniveaux pour sélectionner la feuille à améliorer, et (ii) nous adaptons les transferts de biens entre agents pour maintenir une allocation multiniveaux cohérente. Nous décrivons maintenant **MGYS**, en soulignant les différences avec **GYS**. Nous modifions d'abord \mathcal{T} en ajoutant une feuille supplémentaire, notée 0, dont le parent est 1. Bien que nous ne tenions pas compte du nœud 0 lors de la discussion sur l'efficacité et l'équité, nous le traitons parfois comme un agent ayant pour fonction d'utilité $u_0(S) = |S|$. **MGYS** commence avec l'allocation multiniveaux π qui attribue tous les biens au nœud 0. À chaque itération, il sélectionne une feuille qui peut soit prendre un nouveau bien au nœud 0, soit voler un bien à une autre feuille sous certaines conditions. Le protocole de sélection et l'étape d'allocation sont décrits ci-dessous.

Graphe d'échange et transferts de biens. Le *graphe d'échange* d'une allocation multiniveaux donnée $\pi \in \Pi$, noté $G(\pi)$, est un graphe orienté sur l'ensemble des biens \mathcal{G} . Il y a un arc de $g \in \mathcal{G}$ possédé par $x \in \mathcal{L}$ vers $g' \in \mathcal{G}$ possédé par $y \in \mathcal{L} \setminus \{x\}$ dans $G(\pi)$ si x est indifférent entre g et g' étant donné son lot de biens, c'est-à-dire si $u_x((\pi(x) \setminus \{g\}) \cup \{g'\}) = u_x(\pi(x))$. Étant donné une feuille $x \in \mathcal{L}$, soit $F_x(\pi) = \{g \in \mathcal{G} : \Delta^{u_x}(\pi(x), g) = 1\}$ l'ensemble des biens pour lesquels x a un gain marginal strictement positif. Pour toutes feuilles $x, y \in \mathcal{L}$ avec $x \neq y$, un *chemin de transfert* est un chemin d'un nœud dans $F_x(\pi)$ vers un nœud dans $\pi(y)$. Étant donné un chemin de transfert $T = (g_1, g_2, \dots, g_t)$, nous appelons *augmentation de chemin* l'opération consistant à réallouer le bien g_t au propriétaire de g_{t-1} , g_{t-1} au propriétaire de g_{t-2} , et ainsi de suite, jusqu'à ce que g_1 soit réalloué à la feuille x . Un changement supplémentaire par rapport à **GYS** est que **MGYS** s'assure que π reste une allocation multiniveaux après la réallocation en mettant à jour de manière appropriée les lots des nœuds internes : lorsqu'une feuille z reçoit le bien g d'une autre feuille z' , chaque ancêtre $i \in \text{Anc}(z) \setminus \text{Anc}(z')$ ajoute g à son lot tandis que chaque ancêtre $i \in \text{Anc}(z') \setminus \text{Anc}(z)$ le retire.

Fonctions de gain et sélection des feuilles. Comme dans **GYS**, notre algorithme utilise des fonctions de gain pour guider l'allocation vers des résultats équitables. Alors que **GYS** sélectionne la feuille à améliorer en maximisant simplement la fonction de gain, **MGYS** le fait en utilisant une approche descendante. Pour tout nœud $i \in \mathcal{N} \setminus \{1\}$, soit $\phi_{\mathcal{P}(i)}(\pi, i)$ la *fonction de gain* de l'agent i , capturant l'amélioration marginale due à l'obtention d'un bien sous l'allocation π . Notons que cette fonction dépend de la notion d'équité de $\Psi_{\mathcal{P}(i)} \in \mathcal{F}$, et renvoie un vecteur de dimension b calculé à partir de $v_i(\pi)$. Les fonctions de gain pour les critères d'équité susmentionnés sont fournies dans la version complète du papier. À chaque itération, la feuille x choisie pour amélioration est sélectionnée de manière descendante : en partant de la racine, nous choisissons l'enfant

$i \in \mathcal{C}(1) \setminus \{0\}$ tel que $\phi_1(\pi, i)$ domine lexicographiquement² tous les autres, en départageant les égalités par le plus petit indice. Nous répétons le processus sur les enfants de i , et itérons jusqu'à atteindre une feuille. Cette feuille est ensuite utilisée pour effectuer une augmentation de chemin, si possible. Ce processus est appelé `Select_Leaf`.

Algorithme. MGYS commence avec une allocation multiniveaux $\pi \in \Pi$ où chaque bien est non alloué, c'est-à-dire qu'il appartient au nœud 0. Ensuite, à chaque itération, nous sélectionnons une feuille $x \in \mathcal{L}$ selon le protocole de sélection décrit ci-dessus. Puis, s'il existe un chemin de transfert de $F_x(\pi)$ vers $\pi(0)$, nous trouvons le plus court et effectuons l'augmentation de chemin correspondante. Sinon, nous retirons x de l'arbre puis élaguons itérativement tout ancêtre devenu sans enfant. **MGYS** s'arrête lorsque les seuls nœuds restants sont la racine et le nœud 0. Les pseudo-codes se trouvent dans la version complète du papier.

Exemple 4. Nous exécutons **MGYS** sur l'instance de l'Exemple 3. Rappelons que Ψ_1 est le bien-être de Nash pondéré maximum, donc pour tout $j \in \mathcal{C}(1)$, la fonction de gain est $\phi_1(\pi, j) = (1 + 1/v_j(\pi))^{w_j}$ si $v_j(\pi) > 0$, et arbitrairement grande sinon. Puisque Ψ_2 et Ψ_3 correspondent à la dominance de Lorenz, nous avons $\phi_2(\pi, j) = \phi_3(\pi, j') = -v_j(\pi)$ pour $j \in \mathcal{C}(2)$ et $j' \in \mathcal{C}(3)$. Initialement $\pi(0) = \pi(1) = \mathcal{G}$. À la première étape, la feuille 4 est sélectionnée : le nœud 2 est priorisé à la racine (puisque les égalités sont départagées par le plus petit indice), et parmi ses enfants, la feuille 4 est choisie (par départage d'égalité). Une augmentation de chemin se produit : 4 sélectionne g_1 . Les lots sont modifiés : $\pi(4) = \pi(2) = \{g_1\}$ et $\pi(0) = \{g_2, \dots, g_5\}$. À la deuxième étape, la feuille 6 est sélectionnée : $\phi_1(\pi, 3) > \phi_1(\pi, 2)$, et parmi les enfants de 3, la feuille 6 remporte l'égalité. Une augmentation de chemin se produit : 6 prend g_2 donc $\pi(6) = \pi(3) = \{g_2\}$ et $\pi(0) = \{g_3, g_4, g_5\}$. Ensuite, la feuille 5 est sélectionnée : $\phi_1(\pi, 2) > \phi_1(\pi, 3)$ et $\phi_2(\pi, 5) > \phi_2(\pi, 4)$. Elle choisit g_3 : $\pi(5) = \{g_3\}$, $\pi(2) = \{g_1, g_3\}$, $\pi(0) = \{g_4, g_5\}$. Puis 4 est sélectionnée : $\phi_1(\pi, 2) > \phi_1(\pi, 3)$ et 4 est choisie par départage d'égalité. Aucun chemin de transfert n'existe de 4 vers $\pi(0)$, et donc 4 est retirée de \mathcal{T} . La même chose se produit avec 5. Ensuite, 2 est retiré puisqu'il n'a plus d'enfants. La feuille 7 est sélectionnée (car $\phi_3(\pi, 7) > \phi_3(\pi, 6)$), et prend g_4 : $\pi(7) = \{g_4\}$, $\pi(3) = \{g_2, g_4\}$, $\pi(0) = \{g_5\}$. Ensuite, la feuille 6 est sélectionnée par départage d'égalité et retirée de \mathcal{T} car aucun chemin de transfert n'existe. Enfin, 7 est choisie et sélectionne g_5 ; **MGYS** se termine après avoir retiré les nœuds 7 et 3. Nous obtenons : $\pi(1) = \mathcal{G}$, $\pi(2) = \{g_1, g_3\}$, $\pi(3) = \{g_2, g_4, g_5\}$, $\pi(4) = \{g_1\}$, $\pi(5) = \{g_3\}$, $\pi(6) = \{g_2\}$, $\pi(7) = \{g_4, g_5\}$.

Notons que **MGYS** se termine puisque, à chaque étape, soit des nœuds sont supprimés de \mathcal{T} , soit un bien est retiré de $\pi(0)$ par augmentation de chemin. Si $\pi(0) = \emptyset$, alors il ne reste plus de chemin de transfert, et l'algorithme supprime

2. Given two vectors $x, y \in \mathbb{R}^c$ for some positive integer c , x lexicographically dominates y if there exists $k \in \{1, \dots, c\}$ such that $x_j = y_j$ for all $j \in \{1, \dots, k-1\}$ and $x_k > y_k$.

des nœuds jusqu'à ce que la condition de la boucle while soit remplie. Avant de montrer que **MGYS** renvoie une allocation optimale au sens utilitariste multiniveaux par rapport à v , nous présentons quelques lemmes utiles (preuves dans la version complète du papier).

Lemme 6. L'allocation multiniveaux π est non redondante si et seulement si $v_i(\pi) = |\pi(i)|$ pour tous les nœuds $i \in \mathcal{N}$.

Nous adaptons le Lemme 2.2 et le Lemme 2.3 de [19] à notre cadre multiniveaux :

Lemme 7. Soient $\pi, \pi' \in \Pi$ deux allocations multiniveaux non redondantes. Étant donné une feuille $x \in \mathcal{L}$ telle que $|\pi(x)| < |\pi'(x)|$, il existe un chemin dans $G(\pi)$ de $F_x(\pi)$ vers $\pi(y)$ pour une certaine $y \in \mathcal{L}$ telle que $|\pi(y)| > |\pi'(y)|$.

Lemme 8. Soit $\pi \in \Pi$ une allocation multiniveaux non redondante. Soit $T = (g_1, \dots, g_t)$ l'un des plus courts chemins de $F_x(\pi)$ vers $\pi(y)$ pour certains $x, y \in \mathcal{L}$. Soit $\pi' \in \Pi$ l'allocation multiniveaux obtenue à partir de π en effectuant l'augmentation de chemin correspondant à T . L'allocation multiniveaux π' est non redondante et vérifie pour chaque $i \in \mathcal{N}$:

$$|\pi'(i)| = \begin{cases} |\pi(i)| + 1 & \text{si } i \in \text{Anc}(x) \cup \{x\} \setminus \text{Anc}(y) \\ |\pi(i)| - 1 & \text{si } i \in \text{Anc}(y) \cup \{y\} \setminus \text{Anc}(x) \\ |\pi(i)| & \text{sinon} \end{cases}$$

Lemme 9. L'allocation multiniveaux $\pi \in \Pi$ maintenue par **MGYS** est non redondante à toute itération.

Nous prouvons maintenant que **MGYS** garantit l'optimalité au sens utilitariste multiniveaux par rapport à v , mais n'assure pas l'équité multiniveaux.

Théorème 10. **MGYS** renvoie une allocation optimale au sens utilitariste multiniveaux par rapport à v .

Démonstration. Supposons par l'absurde que π^* n'est pas optimale au sens utilitariste multiniveaux par rapport à v . Soit $\pi' \in \Pi$ une allocation multiniveaux non redondante optimale au sens utilitariste par rapport à v minimisant $\sum_{x \in \mathcal{L}} |u_x(\pi^*(x)) - u_x(\pi'(x))|$. Puisque π^* est non redondante (par le Lemme 9) et n'est pas optimale au sens utilitariste multiniveaux (par hypothèse), $|\pi^*(y)| < |\pi'(y)|$ est vraie pour une certaine $y \in \mathcal{L}$ (par le Lemme 6). Supposons que $|\pi^*(x)| \leq |\pi'(x)|$ est vraie pour toutes les autres feuilles $x \in \mathcal{L} \setminus \{y, 0\}$. Puisque π^*, π' sont non redondantes, nous devons avoir $|\pi^*(0)| > |\pi'(0)|$. Soit π_0 l'allocation multiniveaux au début de l'itération où y est retirée de \mathcal{T} . Notons que π_0 est non redondante par le Lemme 9. Par construction : $|\pi_0(y)| = |\pi^*(y)| < |\pi'(y)|$, $|\pi_0(x)| \leq |\pi^*(x)| \leq |\pi'(x)|$ pour toute $x \in \mathcal{L} \setminus \{0\}$ et $|\pi_0(0)| \geq |\pi^*(0)| > |\pi'(0)|$. Par le Lemme 7, il y a un chemin de transfert de $F_y(\pi_0)$ vers $\pi_0(0)$ dans $G(\pi_0)$, ce qui contredit la définition de π_0 . Par conséquent, nous devons avoir $|\pi^*(y')| > |\pi'(y')|$ pour une certaine feuille $y' \in \mathcal{L} \setminus \{0\}$. Par le Lemme 7, il y a un chemin de transfert de $F_{y'}(\pi')$ vers $\pi'(z)$ dans $G(\pi')$ pour une certaine $z \in \mathcal{L}$ avec $|\pi^*(z)| < |\pi'(z)|$. Par le Lemme 8, transférer des

biens le long de ce chemin conduit à une allocation multiniveaux non redondante π_1 telle que $|\pi_1(y')| = |\pi'(y')| + 1$, $|\pi_1(z)| = |\pi'(z)| - 1$ et $|\pi_1(x')| = |\pi'(x')|$ pour toute $x' \in \mathcal{L} \setminus \{y', z\}$. Si $z = 0$, alors $v_1(\pi_1) > v_1(\pi')$ (puisque y' a augmenté son utilité sans réduire celle d'aucun autre agent), ce qui contredit l'optimalité à la racine de π' . Si $z \neq 0$, alors nous contredisons son hypothèse de minimalité puisque $\sum_{x \in \mathcal{L}} |u_x(\pi^*(x)) - u_x(\pi'(x))| > \sum_{x \in \mathcal{L}} |u_x(\pi^*(x)) - u_x(\pi_1(x))|$. \square

Théorème 11. *MGYS peut échouer à renvoyer une allocation Ψ -maximisante multiniveaux par rapport à v , même lorsque $\Psi_i \in \mathcal{F}$ pour tout $i \in \mathcal{I}$.*

Démonstration. Considérons l'Exemple 3, et soient π_1^* et π_2^* les allocations multiniveaux renvoyées respectivement par **SMA** et **MGYS** (voir Exemples 3-4). Tant $\pi_1^*|_{\mathcal{C}(1)}$ que $\pi_2^*|_{\mathcal{C}(1)}$ appartiennent à $\mathcal{A}_{\mathcal{C}(1)}^G$. Le bien-être de Nash pondéré est de $3^5 \times 2^2 = 972$ pour $\pi_1^*|_{\mathcal{C}(1)}$ et de seulement $2^5 \times 3^2 = 288$ pour $\pi_2^*|_{\mathcal{C}(1)}$. Ainsi $\pi_2^*|_{\mathcal{C}(1)}$ n'est pas Ψ_1 -maximisante par rapport à v . Par conséquent, π_2^* n'est pas Ψ -maximisante multiniveaux par rapport à v . Notons que cette instance constituerait toujours un contre-exemple si Ψ_1 était la dominance de Lorenz, le leximin pondéré ou le bien-être des p -moyennes pondéré (par ex. avec $p = 1$). Des contre-exemples similaires existent dans des cadres non pondérés. \square

Nous énonçons maintenant la complexité temporelle dans le pire des cas de **MGYS** (preuve dans le papier complet). Soit T_u le temps dans le pire des cas pour évaluer l'utilité d'une feuille (c.-à-d. pour une MRF donnée), et soit b la dimension maximale des fonctions de gain ϕ_i pour $i \in \mathcal{I}$. Pour les notions d'équité considérées ici, b est dans $O(1)$.

Théorème 12. *La complexité temporelle dans le pire des cas de MGYS est $O((n + m)(mT_u \log m + mn + nb))$.*

Notons que dans certains cas, **SMA** peut engendrer une complexité de calcul beaucoup plus élevée que **MGYS**. Cela s'explique par le fait que le calcul de $\hat{v}_i(S)$ pour un nœud interne $i \in \mathcal{I}$ et un ensemble $S \subseteq \mathcal{G}$ nécessite de résoudre un problème de partition de matroïde, ce qui reste relativement coûteux. En utilisant l'algorithme de [18], cela prend un temps $O((m_i \log m_i + n_i) \cdot \sqrt{m_i} \cdot T_u)$, où $m_i = |S|$ et $n_i = |\mathcal{L}(i)|$. À titre d'illustration, considérons une instance où $m > n$, et où la racine a deux enfants, et appliquons **SMA** en utilisant **GYS** pour calculer chaque A_i lors des appels récursifs. Le premier appel à la racine nécessite alors d'exécuter **GYS** sur deux agents, où chaque évaluation d'utilité implique de résoudre un problème de partition de matroïde. Cela conduit à une complexité temporelle de l'ordre de $m^{7/2}(\log m)^2 \cdot T_u + n\sqrt{m} \log m \cdot T_u + mb$ rien que pour l'appel initial, ce qui dépasse déjà la complexité globale de **MGYS**.

De plus, nous notons également que **MGYS** est légèrement moins général que **SMA**, car il requiert des notions d'équité pour lesquelles une fonction de gain cohérente peut être définie. Par exemple, définir une telle fonction pour EF1 serait difficile, comme cela a déjà été observé dans [19].

n	m	SMA		MGYS	
		Équilibré	Peigne	Équilibré	Peigne
12	20	3.594	4.850	0.136	0.127
12	50	226.963	377.806	2.323	2.120
12	100	-	-	18.040	16.192
10	20	2.756	4.854	0.134	0.127
20	20	3.734	4.339	0.164	0.193
30	20	4.184	4.537	0.215	0.213
30	60	525.322	736.185	3.632	4.614
40	80	-	-	9.048	9.797
150	300	-	-	553.209	533.791

TABLE 1 – Temps de calcul (en sec.) ($p = 0.5$).

p	Algorithme	Arbre équilibré		Arbre en peigne	
		err_1	err_2	err_1	err_2
0.1	MGYS	0.19	2.47	0.18	2.34
	GYS	0.81	3.43	0.84	11.31
0.5	MGYS	0.00	0.00	0.08	2.80
	GYS	0.97	3.19	1.00	26.66
0.9	MGYS	0.00	0.00	0.00	0.00
	GYS	0.97	3.04	0.97	27.51

TABLE 2 – Erreurs pour les instances ne considérant que la dom. de Lorenz pour $n = 15$, $m = 25$.

5 Tests numériques

Nous comparons maintenant les temps de calcul de **SMA** et **MGYS** et évaluons l'équité de **MGYS** en mesurant la fraction des instances où il échoue à renvoyer une allocation Ψ -maximisante multiniveaux (err_1). Pour ces instances, nous mesurons l'erreur comme la somme, sur tous les agents, des différences absolues entre la taille de leur lot et celle dans une allocation locale Ψ_i -maximisante (err_2). Comme référence de base, nous comparons les performances d'équité de **MGYS** à celles de **GYS** appliqué aux feuilles, en mettant à jour les lots des nœuds internes en conséquence.

La version complète du papier fournit tous les détails sur le protocole (matériel, structures d'arbres, préférences, poids, configuration des algorithmes, paramètres de reproductibilité) ainsi que les résultats numériques complets (moyennes, écarts-types, intervalles de confiance).

Les tableaux 1 et 2 présentent les résultats de temps et d'équité moyennés sur 200 instances, étayant notre analyse. Le tableau 1 montre que **MGYS** surpasse **SMA** à la fois en temps d'exécution et en passage à l'échelle. Pour des arbres équilibrés et un nombre fixe d'agents ($n = 12$), l'augmentation du nombre d'objets (de 20 à 50) fait passer l'avantage de vitesse de **MGYS** sur **SMA** d'un facteur 26 à un facteur 98. Pour un nombre fixe d'objets ($m = 20$), **MGYS** est 20 fois plus rapide que **SMA** (cependant, ce ratio reste constant lorsque n augmente). Notons que les deux algorithmes sont plus sensibles aux augmentations du nombre d'objets qu'aux augmentations du nombre d'agents (comme attendu d'après l'analyse de complexité). De plus, **SMA** atteint la limite de temps de 20 minutes pour des instances supérieures à 12 agents et 50 objets, tandis que **MGYS** traite avec succès des cas beaucoup plus grands (par exemple, $n = 200$, $m = 400$). Le tableau 1 montre également que **MGYS** est robuste à la structure hiérarchique sous-jacente, tandis que **SMA** est moins performant sur les arbres en peigne que sur les arbres équilibrés. Nous observons également que le paramètre de parcimonie des préférences p n'a pas d'impact notable sur le temps de calcul.

Nous examinons maintenant l'équité de **MGYS**. Le ta-

bleau 2 montre que **MGYS** échoue rarement à produire une allocation Ψ -maximisante multiniveaux, avec un err_1 proche de zéro dans la plupart des cas. Même avec des préférences restrictives ($p = 0.1$), où **MGYS** manque la condition d'équité dans 19% des instances, le taux d'erreur reste modéré — en particulier comparé à **GYS** sur les feuilles. De plus, dans ces rares cas, les allocations restent proches de l'équité (car err_2 est très faible). En effet, nous observons que err_2 est légèrement supérieur à 2, ce qui indique qu'en moyenne, l'allocation calculée par **MGYS** diffère d'une allocation équitable multiniveaux d'un seul objet mal attribué (puisque err_2 compte chaque objet mal attribué deux fois). La version complète du papier montre que ces résultats se maintiennent pour différentes tailles d'instances, différentes notions d'équité et ne se dégradent que légèrement pour les arbres en peigne. Cette forte performance en matière d'équité est particulièrement remarquable compte tenu des mauvais résultats obtenus par **GYS** appliqué aux feuilles. En effet, nous observons que non seulement **GYS** échoue presque toujours à calculer une allocation Ψ -maximisante multiniveaux par rapport à v (avec un err_1 très proche de 1), mais il peut aussi être loin d'être équitable, surtout pour les arbres en peigne (err_2 pouvant être très élevé). Ces résultats montrent que calculer une allocation équitable multiniveaux est fondamentalement différent de simplement résoudre une allocation mono-niveau au niveau des feuilles.

Références

- [1] Rediet Abebe, Jon Kleinberg, and David C. Parkes. Fair division via social comparison. In *Proceedings of the 16th Conference on Autonomous Agents and MultiAgent Systems*, 2017.
- [2] Gagan Aggarwal, Marios Mertzaniadis, Alexandros Psomas, and Di Wang. Mechanism design with delegated bidding, 2024.
- [3] Martin Aleksandrov and Toby Walsh. Group envy freeness and group pareto efficiency in fair division with indivisible items. In *KI 2018 : Advances in Artificial Intelligence*.
- [4] Haris Aziz and Simon Rey. Almost group envy-free allocation of indivisible goods and chores. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI'20*, 2021.
- [5] Moshe Babaioff, Tomer Ezra, and Uriel Feige. Fair and truthful mechanisms for dichotomous valuations. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021*.
- [6] Nawal Benabbou, Mithun Chakraborty, Edith Elkind, and Yair Zick. Fairness towards groups of agents in the allocation of indivisible items. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence, 2019*.
- [7] Nawal Benabbou, Mithun Chakraborty, Ayumi Igarashi, and Yair Zick. Finding fair and efficient allocations for matroid rank valuations. *ACM Trans. Economics and Comput.*, 9(4) :21 :1–21 :41, 2021.
- [8] Aurélie Beynier, Yann Chevaleyre, Laurent Gourvès, Ararat Harutyunyan, Julien Lesca, Nicolas Maudet, and Anaëlle Wilczynski. Local envy-freeness in house allocation problems. *Auton. Agents Multi Agent Syst.*, 2019.
- [9] Felix Brandt, Vincent Conitzer, Ulle Endriss, Jérôme Lang, and Ariel D. Procaccia. *Handbook of Computational Social Choice*. Cambridge University Press, USA, 1st edition, 2016.
- [10] Xiaolin Bu, Zihao Li, Shengxin Liu, Jiaxin Song, and Biaoshuai Tao. Fair division with allocator's preference. In *Proceedings of WINE 2023*.
- [11] Ioannis Caragiannis, Nick Gravin, and Xin Huang. Envy-freeness up to any item with high nash welfare : The virtue of donating items. In *Proceedings of the 2019 ACM Conference on Economics and Computation*.
- [12] Deeparnab Chakraborty, Yin Tat Lee, Aaron Sidford, Sahil Singla, and Sam Chiu-wai Wong. Faster matroid intersection. In David Zuckerman, editor, *60th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2019*.
- [13] Mithun Chakraborty, Ayumi Igarashi, Warut Suksompong, and Yair Zick. Weighted envy-freeness in indivisible item allocation. *ACM Trans. Econ. Comput.*, 9(3), August 2021.
- [14] Vincent Conitzer, Rupert Freeman, Nisarg Shah, and Jennifer Wortman Vaughan. Group fairness for the allocation of indivisible goods. In *Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence, 2019*.
- [15] Maria Kyropoulou, Warut Suksompong, and Alexandros A. Voudouris. Almost envy-freeness in group resource allocation. In *Proceedings of IJCAI'2019*.
- [16] Jonathan Scarlett, Nicholas Teh, and 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*, page 2466–2468.
- [17] Ulrike Schmidt-Kraepelin, Warut Suksompong, and Steven Wijaya. On multi-level apportionment. *Theory and Decision*, abs/2511.10000, 2025.
- [18] Tatsuya Terao. Faster matroid partition algorithms. *ACM Trans. Algorithms*, 21(2), January 2025.
- [19] Vignesh Viswanathan and Yair Zick. A general framework for fair allocation under matroid rank valuations. *ACM Trans. Econ. Comput.*, September 2025.