Share-Based and Envy-Based Approaches to Fair Division of Indivisible Goods

A dissertation submitted towards the degree Doctor of Engineering of the Faculty of Mathematics and Computer Science of Saarland University

by Hannaneh Akrami

Saarbrücken / 2024

Day of Colloquium:	13 March 2025
Dean of the Faculty:	Prof. Dr. Roland Speicher
Chair of the Committee:	Prof. Danupon Nanongkai, Ph.D.
Reporters:	Prof. Dr. Kurt Mehlhorn
	Prof. Uriel Feige, Ph.D.
	Prof. Edith Elkind, Ph.D.
Academic Assistant:	Nidhi Rathi, Ph.D.

In the name of "Woman, Life, Freedom"

Abstract

Abstract. The fair allocation of resources among agents with individual preferences is a fundamental problem at the intersection of computer science, social choice theory, and economics. This dissertation examines scenarios where the resources to be allocated are a set of indivisible goods. Two primary categories of fairness concepts are considered: share-based and envy-based criteria. Each category encompasses desirable notions of fairness, each with distinct advantages and limitations.

In the first part of the dissertation, we study a share-based fairness notion known as the maximin share (MMS). Since MMS allocations do not always exist, we study relaxed MMS allocations and establish positive results in various settings. In particular, we establish the existence of $(3/4 + 3/3836)$-MMS allocations for agents with additive valuations, and $(3/13)$-MMS allocations for agents with fractionally subadditive (XOS) valuations. In addition, we consider ordinal approximations of MMS and prove the existence of 1-out-of-$4\lceil n/3 \rceil$ MMS allocations in the additive setting.

The second part of the dissertation focuses on envy-based fairness notions. For indivisible items, the most prominent envy-based criterion is envy-freeness up to any item (EFX). Although the existence of EFX allocations remains open in many general settings, we contribute to this area by proving the existence of EFX allocations for three agents under minimal constraints on their valuations. Furthermore, we establish the existence of relaxed forms of EFX, including epistemic EFX, and approximate EFX with charity.

In the third and final part of this thesis, we move beyond single fairness criteria—whether share-based or envy-based—to establish the existence of allocations that satisfy multiple fairness guarantees. Specifically, we prove the existence of (partial) allocations that are 2/3-MMS and EFX simultaneously. This line of research, while less established, represents a promising direction for future research.

Zusammenfassung. Die faire Aufteilung von Ressourcen zwischen Akteuren mit individuellen Präferenzen ist ein grundlegendes Problem an der Schnittstelle von Informatik, Sozialwahltheorie und Wirtschaftswissenschaften. In dieser Dissertation werden Szenarien untersucht, in denen die zu verteilenden Ressourcen eine Reihe von unteilbaren Gütern sind. Es werden zwei Hauptkategorien von Fairnesskonzepten betrachtet: anteilsbasierte und neidbasierte Kriterien. Jede Kategorie umfasst wünschenswerte Vorstellungen von Fairness, die jeweils ihre eigenen Vorteile und Grenzen haben.

Im ersten Teil dieser Arbeit untersuchen wir einen anteilsbasierten Fairnessbegriff, der als Maximin-Share (MMS) bekannt ist. Da MMS-Zuteilungen nicht immer existieren, untersuchen wir relaxierte MMS-Zuteilungen und zeigen positive Ergebnisse in verschiedenen Situationen. Insbesondere beweisen wir die Existenz von $(3/4 + 3/3836)$-MMS-Zuteilungen für Agenten mit additiven Bewertungen und $(3/13)$-MMS-Zuteilungen für Agenten mit fraktionell subadditiven (XOS) Bewertungen. Zusätzlich betrachten wir ordinale Approximationen von MMS und beweisen die Existenz von 1-aus-$4\lceil n/3 \rceil$ MMS-Zuteilungen in dem additiven Fall.

Der zweite Teil der Dissertation befasst sich mit neidbasierten Fairnesskonzepten. Für unteilbare Güter ist das bekannteste neidbasierte Kriterium die Neidfreiheit bis auf ein beliebiges Gut (EFX). Obwohl die Existenz von EFX-Zuteilungen in vielen allgemeinen Kontexten offen bleibt, leisten wir einen Beitrag zu diesem Gebiet, indem wir die Existenz von EFX-Zuteilungen für drei Agenten unter minimalen Beschränkungen ihrer Bewertungen beweisen. Darüber hinaus etablieren wir die Existenz von relaxierten Formen von EFX, einschließlich epistemischem EFX, und approximiertem EFX mit Spenden.

Im dritten und letzten Teil dieser Arbeit gehen wir über einzelne Fairnesskriterien hinaus — sei es anteilsbasiert oder neidbasiert — und beweisen die Existenz von Zuteilungen, die mehrere Fairnessgarantien erfüllen. Insbesondere beweisen wir die Existenz von (teilweisen) Zuteilungen, die gleichzeitig 2/3-MMS und EFX sind. Dieser Forschungszweig ist zwar weniger etabliert, stellt aber eine vielversprechende Richtung für zukünftige Forschung dar.

Acknowledgments

I want to express my deepest gratitude towards Kurt for being an incredible supervisor. For all the freedom he gave me throughout my PhD which built my professional independence and for all the inspiration I have received from him not only about research but also about life in general. I want to thank Naveen for being my first mentor in research, showing me how fun it is to work with great people like himself, and guiding me towards the path which I am currently in.

I thank Jugal and Ruta for hosting me for a very stimulating research visit and I further thank Jugal for unofficially supervising me through the many projects we had which I greatly enjoyed. I thank Masoud who created the initial sparks of my interest in algorithmic game theory by being an amazing teacher. I thank all my collaborators. It has been a pleasure to work with you and I have learned a lot from you. I thank Bhaskar for being an amazing mentor, a great collaborator, and above all, a true friend.

I express my gratitude towards the reviewers of this thesis, Uriel Feige and Edith Elkind, for devoting their very valuable time.

It has been an amazing experience to be part of the D1 group of MPI for Informatics. I thank all the group members who created such a nice and warm atmosphere. I further thank the IST team, Christina, and Elisa for their constant help.

I thank my friends and partners without whom life would just be utterly senseless. To name a few, thank you Sara, Mehrnaz, Sepideh, Golnoosh, Sepehr, Paolo, Anita, Nidhi, Anton, Jacob, Teuta, Felix, and André for all the joy you have brought into my life, all the support, all the shared interests, all the warm hugs and all the love.

From the bottom of my heart I thank my big family for their love and support. I deeply thank my beloved grandmother *mamama* who unfortunately is not among us anymore to see me graduate. She had an endless thirst for knowledge, a drop of which I have inherited. She always encouraged me about education and was an amazing teacher herself. I thank my big-hearted *khale* (aunt) for her unending kindness. I thank my lovely siblings and siblings-in-law Reyhaneh, Hamed, Amirhossein, Fatemeh, Narges, my dear sweet nephews and niece Aliaria, Amirsadra, Amirtaha, Ariana, my caring *baba*, and above all, my loving *madar*, the hero of my life.

Dedication

Dedicated to my mamama.

Preface

The papers I decided to include in my thesis are on fair allocation of indivisible goods when agents have additive or more general valuation functions. All of the works included in this thesis were created in close collaboration with my co-authors and the contribution is equally shared between the authors of each paper; the ordering of the names is always alphabetical:

H. Akrami, J. Garg, E. Sharma, and S. Taki. Simplification and Improvement of MMS Approximation. In *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence, IJCAI 2023*

H. Akrami, N. Alon, B. R. Chaudhury, J. Garg, K. Mehlhorn, and R. Mehta. EFX: A Simpler Approach and an (Almost) Optimal Guarantee via Rainbow Cycle Number. In *Proceedings of the 24th ACM Conference on Economics and Computation, EC 2023*
The full version of this paper is published in *Operations Research 2025*.

H. Akrami, K. Mehlhorn, M. Seddighin, and G. Shahkarami. Randomized and Deterministic Maximin-share Approximations for Fractionally Subadditive Valuations. *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023*

H. Akrami, J. Garg, E. Sharma, and S. Taki. Improving Approximation Guarantees for Maximin Share. In *Proceedings of the 25th ACM Conference on Economics and Computation, EC 2024*

H. Akrami and J. Garg. Breaking the 3/4 Barrier for Approximate Maximin Share. In *Proceedings of the 2024 ACM-SIAM Symposium on Discrete Algorithms, SODA 2024*

H. Akrami and N. Rathi. Epistemic EFX Allocations Exist for Monotone Valuations. In *Proceedings of the The Thiry-Ninth AAAI Conference on Artificial Intelligence, AAAI 2025*

H. Akrami and N. Rathi. Achieving Maximin Share and EFX/EF1 Guarantees Simultaneously. In *Proceedings of the The Thiry-Ninth AAAI Conference on Artificial Intelligence, AAAI 2025*

Additionally, I published four more papers during my PhD: one on fair allocation of indivisible chores, and three on fair allocation of indivisible goods when agents have valuation functions that are more restricted than additive, such as bi-valued or restricted additive functions:

H. Akrami, B. R. Chaudhury, M. Hoefer, K. Mehlhorn, M. Schmalhofer, G. Shahkarami, G. Varricchio, Q. Vermande, E. van Wijland. Maximizing Nash Social Welfare in 2-Value Instances: Delineating Tractability. *Mathematics of Operations Research (Minor Revision)*

H. Akrami, B. R. Chaudhury, M. Hoefer, K. Mehlhorn, M. Schmalhofer, G. Shahkarami, G. Varricchio, Q. Vermande, E. van Wijland. Maximizing Nash Social Welfare in 2-Value Instances. In *Proceedings of the The Thiry-Sixth AAAI Conference on Artificial Intelligence, AAAI 2022*

H. Akrami, R. Rezvan, M. Seddighin. An EF2X Allocation Protocol for Restricted Additive Valuations. In *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI 2023*

H. Akrami, B. R. Chaudhury, J. Garg, K. Mehlhorn, R. Mehta. Fair and Efficient Allocation of Indivisible Chores with Surplus. In *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence, IJCAI 2023*

Contents

CHAPTER 1

Introduction

Fair division of resources is a fundamental problem in many disciplines, including computer science, economics, operations research, and social choice theory. In a classical fair division problem, the goal is to "fairly" allocate a set of items among a set of agents [64]. Such problems find very early historical mentions, for instance, in ancient Greek mythology and the Bible. Even more so today, many real-life scenarios are paradigmatic of the problems in this domain, e.g., division of family inheritance [60], divorce settlements [18], spectrum allocation [34], air traffic management [70], course allocation [21], and many more.[1]

The field of fair division encompasses a broad range of problems, depending on the properties of the items, the notion of fairness being considered, and the agents' valuations of the items.

The items are considered to be "goods" or "chores". In a very general sense, if the agents are happy to receive an item, that item is considered to be a good and otherwise a chore. Hence, in the mixed setting, it could be the case that an item is a good for some agent and a chore for some other. Moreover, depending on what bundle each agent currently has, a remaining item could be considered a good or a chore. Furthermore, the items can be divisible or indivisible. If an item is divisible, then agents could receive a fraction of that item. In contrast, if an item is indivisible, it should be allocated to at most one agent.

In this dissertation, we focus on important open problems in discrete fair division, where a set $\mathcal{M}$ of m *indivisible goods* needs to be allocated to a set $\mathcal{N}$ of n agents. From now on, we use "items" and "goods" interchangeably. Each agent i is equipped with a valuation function $v_i\colon 2^{\mathcal{M}} \rightarrow \mathbb{R}_{\geq 0}$ which captures the utility agent i derives from any bundle that can be allocated to her. One of the most well studied classes of valuations is the class of *additive valuations*, i.e., $v_i(S) = \sum_{g \in S} v_i(\{g\})$ for all $S \subseteq \mathcal{M}$. However, we also study more general classes of valuation functions like submodular, fractionally subadditive, and general monotone (see Chapter 2 for formal definitions). The goal is to determine a *fair* (sub)partition $X = (X_1, X_2, \ldots, X_n)$ of $\mathcal{M}$ where X_i is allocated to agent i. Depending on the considered notion of fairness, this setting has several different problems.

The literature has long examined different notions of fairness, primarily through envy-based and share-based perspectives. In share-based frameworks, agents are satisfied if their utility exceeds a certain threshold, independent of others' bundles. Conversely, envy-based frameworks involve agents comparing their received bundles to those of others, where feelings of fairness hinge on these comparisons. Notably, an agent may receive a substantial utility but still feel dissatisfied if others receive more.

[1] See `www.spliddit.org` and `www.fairoutcomes.com` for a more detailed explanation of fair division protocols used in day-to-day problems.

1.1 Share-Based Notion

In share-based notions, an agent finds an allocation fair only through the value she obtains from her bundle (irrespective of what others receive). For each agent i, if the value i receives is at least some threshold t_i, then the allocation is said to be fair. *Proportionality* is an important share-based notion of fairness that entails an allocation to be fair when every agent $i \in \mathcal{N}$ values her bundle at least as much as her proportional share of $v_i(\mathcal{M})/n$ (i.e., $t_i = v_i(\mathcal{M})/n$). It is easy to see that proportionality is too strong to be satisfied in the discrete setting. As a counter-example, consider two agents and one good with a positive utility to both of the agents. Note that no matter how we allocate this good, one agent receives 0 utility, which rules out the existence of proportional allocations and any multiplicative approximation of proportionality. This necessitates studying relaxed fairness notions when goods are indivisible.

The most prominent relaxation of proportionality, and our focus in the first part of this dissertation, is *maximin share (MMS)*, introduced by Budish [20]. MMS is also preferred by participating agents over other notions, as shown in real-life experiments [43]. The maximin share of an agent is the maximum value she can guarantee to obtain if she divides the goods into n bundles (one for each agent) and receives a bundle with the minimum value. Basically, for an agent i, assuming that all agents have i's valuation function, the maximum value one can guarantee for all the agents is i's maximin share, denoted by MMS_i. Formally, for a set S of goods and any positive integer d, let $\Pi_d(S)$ denote the set of all partitions of S into d bundles. Then,

$$\text{MMS}_i^d(S) := \max_{P \in \Pi_d(S)} \min_{j=1}^{d} v_i(P_j).$$

For all agents i, $\text{MMS}_i = \text{MMS}_i^n(\mathcal{M})$. An allocation is MMS, if all agents value their bundles at least as much as their MMS values. Formally, allocation A is MMS, if $v_i(A_i) \geq \text{MMS}_i$ for all agents $i \in \mathcal{N}$.

However, MMS is an unfeasible share guarantee that cannot always be satisfied when there are more than two agents with additive valuations [37, 52, 62]. Therefore, the MMS share guarantee needs to be relaxed, and the two natural ways are its multiplicative and ordinal approximations.

1.1.1 Multiplicative Approximations of MMS

Since we need to lower the share threshold, a natural way is to consider $\alpha < 1$ times the MMS value. Formally, an allocation $X = (X_1, \dots, X_n)$ is α-MMS if for each agent i, $v_i(X_i) \geq \alpha \cdot \text{MMS}_i$. When agents have additive valuations, earlier works showed the existence of 2/3-MMS allocations using several different approaches [6, 13, 41, 52, 62]. Later, in a groundbreaking work, the existence of 3/4-MMS allocations was obtained through more sophisticated techniques and involved analysis [44].

After Ghodsi et al. [44] proved the existence of 3/4-MMS allocations and gave a PTAS to compute one, Garg and Taki [42] gave a simple algorithm with complicated analysis proving the existence of $(\frac{3}{4} + \frac{1}{12n})$-MMS allocations and also computing a 3/4-MMS allocation in polynomial time. The approximation factor of $3/4 + O(1/n)$ is tight for this algorithm [11, 32].

	Existence	**Non-existence**
$n = 3$	$^{11}/_{12}$ [36]	$> {}^{39}/_{40}$ [37]
$n = 4$	$^{4}/_{5}$ [44]	$> 1 - 4^{-4}$ [37]
$n > 4$	$^{2}/_{3}$ [6, 41, 53, 62] $^{2}/_{3}(1 + {}^{1}/_{3n-1})$ [13] $^{3}/_{4}$ [44] $^{3}/_{4} + {}^{1}/_{12n}$ [42] $^{3}/_{4} + {}^{3}/_{3836}$ **(Theorem 3.61)**	$> 1 - \frac{1}{n^4}$ [37]

Table 1.1: Summary of the approximate MMS results when agents have additive valuations

The complementary problem is to find upper bounds on the largest α for which α-MMS allocations exist. Feige, Sapir, and Tauber [37] constructed an example with three agents and nine goods for which no allocation is better than 39/40-MMS. For $n \geq 4$, their construction gives an example for which no allocation is better than $(1-n^{-4})$-MMS. Table 1.1 summarizes all these results. We note that most of these existence results can be easily converted into PTAS for finding such an allocation using the PTAS for finding the MMS values [71].

In Chapter 3, we study the additive setting. First in Section 3.3, we simplify the analysis of (a slight modification of) the Garg-Taki [42] algorithm significantly. Then, by combining novel ideas with existing techniques, in Section 3.4 we prove the existence of $(\frac{3}{4} + \frac{3}{3836})$-MMS allocations, breaking the 3/4 barrier that existed since the work of Ghodsi et al. [44].

So far, we only discussed guarantees for the setting in which agents have additive valuations. Nevertheless, several studies in recent years show approximation guarantees of maximin-share to all agents for other classes of valuation functions, including submodular, fractionally subadditive, and subadditive. We refer to Chapter 2 for a formal definition of these valuation classes. See Table 1.2 for the state-of-the-art guarantees on the maximin-share. For submodular valuation class, a 10/27-MMS allocation can be computed in polynomial time. For all other classes of valuation functions mentioned in Table 1.2, the states-of-the-art are existential results.

Valuation Class	Approximation Guarantee	Upper bound
Additive	$3/4 + 1/(12n)$ [42] $3/4 + 3/3836$ (Theorem 3.61)	$1 - 1/n^4$ [37]
Submodular	$10/27$[68]	$3/4$ [44]
Fractionally Subadditive	0.219225 [63] $3/13$ (Theorem 4.3)	$1/2$ [44]
Subadditive	$1/(\log n \log\log n)$ [63]	$1/2$ [44]

Table 1.2: A summary of the state-of-the-art results for MMS in different valuation classes.

Let us revisit the instance with one item and two agents. Suppose that the item has value 6 for both agents. By definition, the proportional share of each agent is $6/2 = 3$, and since one agent receives no item, satisfying proportionality or any approximation of it is impossible. On the other hand, we have $\mathrm{MMS}_1 = \mathrm{MMS}_2 = 0$. Thus, allocating the item to any agent satisfies maximin-share. Indeed, we can circumvent the non-guaranteed existence of fair allocation by reducing our expectation of fairness to maximin-share. However, regardless of how we allocate the item, one agent receives one item, and the other receives nothing. Therefore, having one agent with zero utility is inevitable for any deterministic allocation in this example. The question then arises: can we do better?

One way to improve the allocation is to use randomization and obtain a better guarantee in expectation (*ex-ante*). A randomized allocation $\mathcal{R}$ has a property $\mathcal{P}$ (e.g. proportionality, α-MMS, etc) ex-ante, if $\mathcal{P}$ holds in expectation. For example, $\mathcal{R}$ is α-MMS ex-ante, if the expected utility of all agents is at least α times their MMS value. Furthermore, randomized allocation $\mathcal{R}$ has a property $\mathcal{P}$ ex-post, if $\mathcal{P}$ holds for all the deterministic allocations in the support of $\mathcal{R}$.

In order to obtain better guarantees in expectation, we can allocate the item to each agent with probability $1/2$. This way, the expected utility of each agent is equal to 3. In economic terms, this allocation satisfies proportionality *ex-ante*. Note that one agent receives no item *ex-post* (that is, after fixing the outcome); however, it guarantees proportionality ex-ante to both agents.

Considering random allocations and ex-ante fairness makes the problem much handier. For instance, assuming there are n items and n agents, allocating each item to each agent with probability $1/n$ satisfies proportionality ex-ante. However, this randomized allocation has no ex-post fairness guarantee: with a non-zero probability, the outcome allocates all the items to one agent, and the rest of the agents receive no item. It is tempting to find allocations that simultaneously admit ex-ante and ex-post guarantees. The support of such an allocation is limited to outcomes with some desirable fairness guarantee. For example, consider the following random allocation: we choose a random permutation of these n items and allocate the i^{th} item in the permutation to agent i. This allocation satisfies proportionality ex-ante and maximin-share ex-post.

Recently, several studies have investigated randomized allocations with both ex-ante and ex-post guarantees, also known as best-of-both-worlds guarantees. Some notable results with the focus on additive valuations are (i) an ex-ante envy-free and ex-post EF1 allocation algorithm [39] and (ii) an ex-ante proportional and ex-post 1/2-MMS allocation algorithm [12]. Recently, Feldman et al. [38] studied best-of-both-worlds for subadditive valuations and gave an allocation algorithm with ex-ante guarantee of 1/2-envy-freeness and ex-post guarantee of 1/2-EFX and EF1.

In Chapter 4, we explore fair deterministic and randomized allocations for fractionally subadditive valuation functions. A valuation function $v_i(\cdot)$ is fractionally subadditive (XOS), if there exists a set of several additive valuation functions $u_{i,1}, u_{i,2}, \ldots, u_{i,\ell} : 2^{\mathcal{M}} \to \mathbb{R}_{\geq 0}$ such that for every set S we have $v_i(S) = \max_{1 \leq k \leq \ell} u_{i,k}(S)$. Fractionally subadditive is a super-class of different set functions such as additive, gross substitute, and submodular. In addition, fractionally subadditive is a subclass of subadditive set functions.

Indeed, we are looking for allocations that satisfy an approximation of maximin-share both ex-ante and ex-post for fractionally subadditive valuations. Of course, we expect our ex-ante guarantee to be stronger than the ex-post one. In Theorem 4.2, we prove the existence of randomized allocations which are 1/4-MMS ex-ante and 1/8-MMS ex-post.

Moreover, we improve the ex-post approximation guarantee (deterministically) for this class of valuations to 3/13 (Theorem 4.3).

1.1.2 Ordinal Approximations of MMS

Another natural way of relaxing MMS is to consider the share value of $\text{MMS}_i^d(M)$ for $d > n$ for each agent i, which is the maximum value that i can guarantee if she divides the goods into d bundles and then takes a bundle with the minimum value. This notion was introduced together with the MMS notion by Budish [20], which also shows the existence of 1-out-of-$(n+1)$ MMS after *adding excess goods*. Unlike α-MMS, this notion is robust to small perturbations in the values of goods because it only depends on the bundles' ordinal ranking and is not affected by small perturbations as long as the ordinal ranking of the bundles does not change.

The α-MMS is very sensitive to agents' precise cardinal valuations: consider the example mentioned by Hosseini, Searns, and and Segal-Halevi [47]. Assume $n = 3$ and there are four goods g_1, g_2, g_3 and g_4 with values 30, 39, 40 and 41 respectively for agent 1. Assume the goal is to guarantee the 3/4-MMS value of each agent. We have $\text{MMS}_1 = 40$, and therefore any non-empty bundle satisfies 3/4-MMS for agent 1. However, if the value of g_3 gets slightly perturbed and becomes $40+\varepsilon$ for any $\varepsilon > 0$, then $\text{MMS}_1 > 40$ and then $3/4 \cdot \text{MMS}_1 > 30$ and the bundle $\{g_1\}$ does not satisfy agent 1. Thus, the acceptability of a bundle (in this example, $\{g_1\}$) might be affected by an arbitrarily small perturbation in the value of an irrelevant good (i.e., g_3). Observe that in this example, whether the value of g_3 is 40 or $40 + \varepsilon$ for any $\varepsilon \in \mathbb{R}$, $\{g_1\}$ is an acceptable 1-out-of-4 MMS bundle for agent 1.

Another way to interpret 1-out-of-d MMS allocations is giving n/d fraction of agents their MMS value and nothing to the remaining agents. Note that since we know MMS allocations do not necessarily exist, meaning that we cannot give all agents their MMS value, it is a valid question to ask how many of the agents we can indeed satisfy up to their MMS value. Apart from the theoretical significance of the question, in practice this scenario can be desirable when we want to favour an under-represented group of agents. Or when the act of allocating resources is a repetitive action and we can favour the set of agents who have not receive anything yet, in later rounds.

In the standard setting (i.e., without excess goods), the first non-trivial ordinal approximation was the existence of 1-out-of-$(2n-2)$ MMS allocations [1], which was later improved to 1-out-of-$\lceil 3n/2 \rceil$ [46], and then to the current state-of-the-art 1-out-of-$\lfloor 3n/2 \rfloor$ [47]. On the other hand, the (non-)existence of 1-out-of-$(n+1)$ MMS allocations is open to date.

Our result in Chapter 5 shows that 1-out-of-$4\lceil n/3 \rceil$ MMS allocations always exist, thereby improving the state-of-the-art of 1-out-of-d MMS.

1.2 Envy-Based Notions

In the realm of envy-based fairness criteria, envy-freeness is the most prominent standard. It requires that no agent prefers another agent's bundle over her own. Like proportionality, envy-freeness is feasible with divisible resources [18, 66, 67] but not with indivisible ones. The same example of two agents and one item with positive value to both of them serves as a counter example for envy-freeness in the discrete setting as well. The concept of envy-freeness up to any item (EFX) serves as a practical relaxation of envy-freeness for indivisible items. Under EFX, any envy an agent i has toward another agent j must be eliminated if any item is removed from agent j's bundle [25]. Formally, an allocation X is EFX if and only if for all agents i and j, $v_i(X_i) \geq v_i(X_j \setminus \{g\})$ for all $g \in X_j$. EFX is, in fact, considered to be the "closest analog of envy-freeness" in discrete fair division [23]. Unfortunately, the existence of EFX allocations is still unresolved despite significant efforts by several researchers [25, 58] and is considered one of the most important open problems in fair division [61]. There have been studies on:

- the existence of EFX allocations in restricted settings;
- the existence of relaxations of EFX allocations.

In the second part of this dissertation, we improve our understanding in both of these settings in a systematic direction towards solving the big problem.

1.2.1 Restricted Settings

Special Valuation. One way of restricting the setting is having restrictive assumptions on the valuation functions of the agents. Plaut and Roughgarden [59] prove EFX allocations exist when all agents have identical (monotone) valuations. They also prove EFX allocations exist and can be computed in polynomial time if all the valuations are additive and all agents have an identical ordering of goods based on their values. Furthermore, the existence of EFX allocations are known when agents have binary [10] or bi-valued [4] valuations.

Small Number of Agents. EFX existence has also been studied when there are a small number of agents. In particular, Plaut and Roughgarden [59] proved the existence of EFX allocations for two agents with monotone valuations. For the case of three agents, the existence of EFX was first shown with additive valuations [27] and then extended to cancelable valuations [15]. In Chapter 6, we simplify and improve this result by showing the existence of EFX allocations when two of the agents have general monotone valuations and one has MMS-feasible valuation (a strict generalization of cancelable valuation functions). Our approach is significantly simpler than the previous ones, which also avoids using the standard concepts of envy-graph and champion-graph and may find use in other fair-division problems.

1.2.2 Relaxations of EFX Allocations

Envy-freeness up to One Item (EF1). A weaker notion of fairness is envy-freeness up to one item or EF1 which requires that any envy from any agent to another must be eliminated by the removal of *some* (instead of *any* in the definition of EFX) good from the envied bundle. Formally, allocation X is EF1, if and only if for all agents i, j, either $v_i(X_i) \geq v_i(X_j)$, or there exists a good $g \in X_j$ such that $v_i(X_i) \geq v_i(X_j \setminus \{g\})$. Although EF1 is is a relaxation of EFX, it has been introduced before EFX [20]. For additive valuations, Caragiannis et al. [24] prove that a simple round-robin algorithm outputs an EF1 allocation. Lipton et al. [55] introduced the technique of envy-cycle elimination which by itself outputs EF1 allocations when agents have monotone valuations and is also used as a subroutine of many subsequent works [7, 15, 30].

Approximate EFX. Similar to MMS, one natural way to relax EFX is to consider approximations of EFX. Formally, an allocation X is α-EFX if and only if $v_i(X_i) \geq \alpha \cdot v_i(X_j \setminus \{g\})$ for all $g \in X_j$. Setting $\alpha = 1$, we get the EFX definition and for $\alpha < 1$, α-EFX is clearly weaker than EFX itself. While the existence of 1-EFX allocations remains open, studies has been done to improve the approximation factor α for which α-EFX allocations exist. Plaut and Roughgarden [59] gave the first such algorithm which admitted a 1/2-approximation in pseudopolynomial time for subadditive valuations, with an extension to polynomial time due to Chan et al. [26]. The approximation ratio for the additive case was further improved by Amanatidis et al. [7] and Farhadi et al. [35] to $\varphi - 1 \approx 0.618$ by combining a round-robin and envy-cycle elimination procedure.

EFX with Charity. Caragiannis, Gravin, and Huang [23] introduced the notion of EFX with charity. Here the goal is to find a fair allocation of a subset of all the goods also known as partial allocations. The term charity is referring to the set of unallocated goods. On an extreme end, we can allocate no item to the agents and give everything to the charity. Then the allocation is envy-free and thus satisfies EFX property and all relaxations of it. However, clearly this is not a desired allocation. So when having partial allocations, in addition to EFX, we need to satisfy some other fairness or efficiency notions to make sure that the allocation is not too wasteful. Caragiannis, Gravin, and Huang [23] show that there always exists a partial EFX allocation X such that for each agent i, we have $v_i(X_i) \geq 1/2 \cdot v_i(X_i^*)$, where $X^* = (X_1^*, X_2^*, \ldots, X_n^*)$ is an allocation with maximum *Nash welfare*. The Nash welfare of an allocation Y is the geometric mean of agents' valuations, $\left(\prod_{i\in[n]} v_i(Y_i)\right)^{1/n}$. It is often considered a direct measure of the fairness and efficiency of an allocation. Following the same line of work, Chaudhury et al. [30] showed the existence of a partial EFX allocation X such that $v_i(X_i) \geq 1/2 \cdot v_i(X_i^*)$ for all agents i, no agent envies the set of unallocated goods, and the total number of unallocated goods is at most $n - 1$.

Approximate EFX with Charity. Chaudhury et al. [28] combined the last two frameworks and studied approximate EFX with charity. In particular, they showed the existence of a $(1-\varepsilon)$-EFX allocation with $O((n/\varepsilon)^{4/5})$ charity for any $\varepsilon > 0$. While the last result is not a strict improvement of the result in [30] (since it ensures $(1-\varepsilon)$-EFX instead of exact EFX), it is the best relaxation of EFX that we can compute in polynomial time, as the algorithm in [30] can only be modified to give $(1-\varepsilon)$-EFX with $n-1$ charity in polynomial time. Another key aspect of the technique in [28] is the reduction of the problem of improving the bounds on charity to a purely graph-theoretic problem. In particular, Chaudhury et al. [28] define the notion of a *rainbow cycle number* $R(d)$ which we formally define in Chapter 7. They prove, the smaller the upper bound on $R(d)$, the lower the number of unallocated goods. They prove $R(d) \in O(d^4)$ and thus establish the existence of $(1-\varepsilon)$-EFX allocation with $O((n/\varepsilon)^{4/5})$ charity. An upper bound of $O(d^2 2^{(\log\log d)^2})$ was obtained by Berendsohn, Boyadzhiyska, and Kozma [14], thereby showing the existence of EFX allocations with $O((n/\varepsilon)^{0.67})$ charity. In Chapter 7, we close this line of improvements by proving an almost tight upper bound on d (matching the lower bound up to a log factor).

Epistemic EFX (EEFX). A recent work of Caragiannis et al. [22] introduced a promising relaxation of EFX, called *epistemic* EFX which adapts the concepts of *epistemic envy-freeness* defined by Aziz et al. [9]. We call an allocation X EEFX if for every agent $i \in [n]$, there exists an allocation Y such that $Y_i = X_i$ and for every bundle $Y_j \in Y$, we have $v_i(X_i) \geq v_i(Y_j \setminus \{g\})$ for every $g \in Y_j$. That is, an allocation is EEFX if, for every agent, it is possible to shuffle the items in the others' bundles so that she becomes "EFX-satisfied". Caragiannis et al. [22] establish existence and polynomial-time computability of EEFX allocations for an arbitrary number of agents with *additive* valuations. We improve this result by proving the existence of EEFX allocations for an arbitrary number of agents with *monotone* valuations. See Theorem 8.7. Note that in the setting of fair division of "goods", monotonicity is the most general assumption one can have for the valuations. Furthermore, we prove that even when agents have identical submodular valuations, the problem of finding EEFX allocations is PLS-hard and requires exponentially many value queries.

1.3 Simultanious Fairness Guarantees

As discussed ealier, in discrete resource-allocation scenarios, the maximin share (MMS) and envy-freeness up to any good (EFX) concepts serve as key representatives of these fairness categories, each addressing distinct fairness aspects. Either of EF1/EFX or MMS properties does not necessarily imply particularly strong approximation guarantees for the other(s) [5]. In Section 9.2, we discuss the guarantees EFX/EF1 allocations can provide for MMS and vice versa. This is in complete contrast to the divisible setting guarantees, where any envy-free allocation is necessarily proportional as well. Hence, it becomes compelling to ask for allocations that attain good guarantees with respect to envy-based and share-based notions of fairness simultaneously. There are few works along these lines in the literature with the assumption that the valuation functions of the agents are additive, e.g., Chaudhury et al. [30] develop a pseudo-polynomial time algorithm to compute a partial allocation that is both 1/2-MMS and EFX. Also, Amanatidis et al. [7] develop an efficient algorithm to compute a complete allocation that is simultaneously 0.553-MMS and 0.618-EFX.

Following this line of research, in Chapter 9, we study fair division instances with agents having additive valuations with the aim of pushing our understanding of the compatibility between two different classes of fairness notions: EFX/EF1 with MMS guarantees. Our main contribution is developing (simple) algorithms which prove the existence of

(1) a partial allocation that is both 2/3-MMS and EFX;

(2) a complete allocation that is both 2/3-MMS and EF1.

If we relax 2/3-MMS to $(2/3 - \varepsilon)$-MMS for any arbitrary constant $\varepsilon > 0$, then the above allocations can be computed in pseudo-polynomial time. If in addition to that, we relax EFX/EF1 to $(1-\delta)$-EFX/$(1-\delta)$-EF1 for any constant $\delta > 0$, then the allocations can be computed in polynomial time.

We note that the above results have led to a new approach for finding desired partial EFX allocations, in particular, where we have a good lower bound on the amount of value each agent receives. It is known that EFX is not compatible with the economic efficiency notion of Pareto optimality [59]. Therefore, it may seem that, in order to guarantee EFX, one might have to sacrifice a lot of utility and agents may not receive bundles with high valuations. Nevertheless, we prove that we can still guarantee their 2/3-MMS value to every agent while finding a partial EFX allocation.

This line of inquiry, while consisting of less established results, represents a novel and promising direction for future research. By investigating allocations that combine both envy-based and share-based guarantees, we aim to push the boundaries of what can be achieved in fair division, particularly in complex scenarios where single criteria may be insufficient.

Bibliographic Notes.

The contributions presented in this thesis are based on the following publications and drafts:

Chapter 3: H. Akrami, J. Garg, E. Sharma, and S. Taki. Simplification and Improvement of MMS Approximation. In *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence, IJCAI 2023*

and

H. Akrami and J. Garg. Breaking the 3/4 Barrier for Approximate Maximin Share. In *Proceedings of the 2024 ACM-SIAM Symposium on Discrete Algorithms, SODA 2024*

Chapter 4: H. Akrami, K. Mehlhorn, M. Seddighin, and G. Shahkarami. Randomized and Deterministic Maximin-share Approximations for Fractionally Subadditive Valuations. *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023*

Chapter 5: H. Akrami, J. Garg, E. Sharma, and S. Taki. Improving Approximation Guarantees for Maximin Share. In *Proceedings of the 25th ACM Conference on Economics and Computation, EC 2024*

Chapter 6 and 7: H. Akrami, N. Alon, B. R. Chaudhury, J. Garg, K. Mehlhorn, and R. Mehta. EFX: A Simpler Approach and an (Almost) Optimal Guarantee via Rainbow Cycle Number. In *Proceedings of the 24th ACM Conference on Economics and Computation, EC 2023*

The full version of this paper is published in *Operations Research 2025.*

Chapter 8: H. Akrami and N. Rathi. Epistemic EFX Allocations Exist for Monotone Valuations. In *Proceedings of the The Thiry-Ninth AAAI Conference on Artificial Intelligence, AAAI 2025*

Chapter 9: H. Akrami and N. Rathi. Achieving Maximin Share and EFX/EF1 Guarantees Simultaneously. In *Proceedings of the The Thiry-Ninth AAAI Conference on Artificial Intelligence, AAAI 2025*

CHAPTER 2

Notation and Preliminaries

For any non-negative integer n, let $[n] = \{1, 2, \ldots, n\}$. A discrete fair division instance $\mathcal{I} = (\mathcal{N}, \mathcal{M}, \mathcal{V})$ consist of a set $\mathcal{N} = [n]$ of n agents, a set $\mathcal{M}$ of m indivisible items and a vector of valuation functions $\mathcal{V} = (v_1, v_2, \ldots, v_n)$ such that for all $i \in [n]$, $v_i : 2^{\mathcal{M}} \to \mathbb{R}$ indicates how much agent i likes each subset of the items. In this dissertation, we consider fair division of indivisible goods, and thus have $v_i(S) \geq 0$ for all $S \subseteq \mathcal{M}$. Also in this case, we either assume $\mathcal{M} = [m]$ or $\mathcal{M} = \{g_1, g_2, \ldots, g_m\}$. For ease of notation, for all $g \in \mathcal{M}$, we sometimes use g instead of $\{g\}$.

Valuation Functions. A set function $f : 2^{\mathcal{M}} \to \mathbb{R}$ is

- *additive*, if $f(S) + f(T) = f(S \cup T) + f(S \cap T)$ for all $S, T \subseteq \mathcal{M}$.
- *submodular*, if $f(S) + f(T) \geq f(S \cup T) + f(S \cap T)$ for all $S, T \subseteq \mathcal{M}$.
- *fractionally subadditive* (XOS), if there exists additive functions $f_1, \ldots, f_\ell : 2^{\mathcal{M}} \to \mathbb{R}$ such that $f(S) = \max_{i \in \ell} f_i(S)$ for all $S \subseteq \mathcal{M}$.
- *subadditive*, if $f(S) + f(T) \geq f(S \cup T)$ for all $S, T \subseteq \mathcal{M}$.
- *increasingly monotone*, if for all $S \subseteq \mathcal{M}$ and $T \subset S$, $f(S) \geq f(T)$.
- *decreasingly monotone*, if for all $S \subseteq \mathcal{M}$ and $T \subset S$, $f(S) \leq f(T)$.

From now on, for fair division of indivisible goods instead of "increasingly monotone" we use "monotone". Throughout the dissertation, we assume that for all valuation functions v, $v(\emptyset) = 0$. In each section, we mention what further assumptions we have on the valuation functions. Let $\mathcal{C}_{\text{additive}}$ be the class of all additive set functions and let $\mathcal{C}_{\text{arg}}$ be defined the same way for arg $\in$ {submodular, XOS, subadditive, monotone}. The following proposition shows the relation between the mentioned classes of set functions.

Proposition 2.1. $\mathcal{C}_{additive} \subset \mathcal{C}_{submodular} \subset \mathcal{C}_{XOS} \subset \mathcal{C}_{subadditive} \subset \mathcal{C}_{monotone}$.

An allocation $X = (X_1, X_2, \ldots, X_n)$ is a partition of a subset of $\mathcal{M}$ into n bundles, such that for all $i \in [n]$, agent i receives bundle X_i. If $\bigcup_{i \in [n]} X_i = \mathcal{M}$, then X is a *complete* allocation and otherwise it is a *partial* allocation. For a (partial) allocation X, we denote the set (pool) of unallocated goods by $P(X)$.

The Nash welfare of an allocation is the geometric mean of agents' utilities under that allocation. It is often considered a direct measure of the fairness and efficiency of an allocation.

Definition 2.2 (Nash Welfare)**.** *The Nash Welfare of an allocation X is*

$$NW(X) = \prod_{i \in [n]} v_i(X_i)^{1/n}.$$

2.1 Share-Based Notions

In share-based class of fairness notions, each agent i has a certain threshold t_i and deems an allocation X fair, if and only if $v_i(X_i) \geq t_i$. An allocation is fair if all agents find it fair. In fair division of divisible goods, the most celebrated share-based notion is *proportionality.*

Definition 2.3 (Proportionality)**.** *Given a fair division instance $\mathcal{I} = ([n], \mathcal{M}, \mathcal{V})$, an allocation X is proportional, if and only if for all agents i, $v_i(X_i) \geq v_i(\mathcal{M})/n$. We define $\pi_i = v_i(\mathcal{M})/n$ as the proportional share of agent i. Similarly, an allocation X is α-proportional for a given $\alpha \geq 0$, if and only if $v_i(X_i) \geq \alpha \cdot \pi_i$.*

In the setting consisting of indivisible items, proportional allocations do not always exist. A simple counter example consists of two agents with positive value over one item: $\mathcal{I} = ([2], \{g\}, (v_1, v_2))$ such that $v_i(g) > 0$ for $i \in [2]$. Although the proportional share $(v_i(g)/2)$ is positive for both of the agents, one agent receives 0 value in any allocation.

Given the infeasibility of proportionality in the setting consisting of indivisible goods, relaxation of proportionality has been introduced. One of the most important ones is *maximin share (MMS)* introduced by Budish [20].

For a set S of goods and any positive integers d, let $\Pi_d(S)$ denote the set of all partitions of S into d bundles. Then,

$$\text{MMS}_{v_i}^d(S) := \max_{P \in \Pi_d(S)} \min_{j=1}^{d} v_i(P_j). \tag{2.1}$$

When clear from the context, we use $\text{MMS}_i^d(S)$ instead of $\text{MMS}_{v_i}^d(S)$. Setting $d = n$ and $S = \mathcal{M}$, we obtain the standard MMS notion. For a given instance $\mathcal{I}$, we abuse the notation and write $\text{MMS}_i = \text{MMS}_i(\mathcal{I}) = \text{MMS}_{v_i}^n(\mathcal{M})$.

Observation 2.4. *Given an instance $\mathcal{I}$ with additive valuations, $\pi_i \geq \text{MMS}_i$ for all agents i.*

Definition 2.5 (α-MMS)**.** *For any $\alpha \geq 0$, an allocation X is α-MMS, if for all agents i, $v_i(X_i) \geq \alpha \cdot \text{MMS}_i$. When $\alpha = 1$, we say allocation X is an MMS allocation.*

Definition 2.6 (1-out-of-d MMS)**.** *For any positive integer d, an allocation X is 1-out-of-d MMS if for all agents i, $v_i(X_i) \geq \text{MMS}_i^d(\mathcal{M})$. When $d = n$, the allocation X is an MMS allocation.*

For each agent i, d-MMS partition of i is a partition $P = (P_1, \dots, P_d)$ of $\mathcal{M}$ into d bundles such that $\min_{j=1}^d v_i(P_j)$ is maximized. Hence, for a d-MMS partition P of agent i, $\text{MMS}_i^d(\mathcal{M}) = \min_{j=1}^d v_i(P_j)$. When $d = n$, we write MMS partition instead of n-MMS partition.

Definition 2.7 (α-MMS preserving)**.** *Given $\alpha > 0$, we say a procedure which takes a fair division instance $\mathcal{I}$ and outputs (possibly) another fair division instance $\widehat{\mathcal{I}}$ is α-MMS-preserving, if given any α-MMS allocation $\widehat{X}$ for $\widehat{\mathcal{I}}$, an α-MMS allocation X for $\mathcal{I}$ can be computed in polynomial time.*

for $i \in \mathcal{N}$ **do**
  if $\text{MMS}_i^d = 0$ **then**
    $\mathcal{N} \leftarrow \mathcal{N} \setminus \{i\}$
  else
    Compute agent i's d-MMS partition $P^{(i)}$.
    $\forall j \in [d]$, $\forall g \in P_j^{(i)}$, let $\widehat{v}_i(g) := v_i(g)/v_i(P_j^{(i)})$.
return $(\mathcal{N}, \mathcal{M}, \widehat{\mathcal{V}})$.

Algorithm 1: `normalize`$(d, (\mathcal{N}, \mathcal{M}, \mathcal{V}))$

Definition 2.8 (d-normalized instance)**.** *An instance $\mathcal{I} = (\mathcal{N}, \mathcal{M}, \mathcal{V})$ is d-normalized, if for all agents i, there exists a partition $P^{(i)} = (P_1^{(i)}, \dots, P_d^{(i)})$ of $\mathcal{M}$ into d bundles such that for all $j \in [d]$, $v_i(P_j^{(i)}) = 1$.*

Note that for a d-normalized instance, every agent's MMS^d value is 1. Furthermore, for each agent i with additive valuation v_i, and for every d-MMS partition Q of agent i, we have $v_i(Q_j) = 1 \; \forall j \in [d]$, since each partition has total value of 1 and $\sum_{j \in [d]} v_i(Q_j) = v_i(\mathcal{M}) = \sum_{j \in [d]} v_i(P_j^{(i)}) = d$.

Algorithm 1 converts a fair division instance with additive valuations to a d-normalized instance.

Lemma 2.9. *Let $(\widehat{\mathcal{N}}, \mathcal{M}, \widehat{\mathcal{V}}) =$ `normalize`$(d, (\mathcal{N}, \mathcal{M}, \mathcal{V}))$ with additive valuations. Then for any allocation A, $v_i(A_i) \geq \widehat{v}_i(A_i) \cdot \text{MMS}_{v_i}^d(\mathcal{M})$ for all $i \in \mathcal{N}$.*

Proof. If $\text{MMS}_{v_i}^d(\mathcal{M}) = 0$, the claim clearly holds for agent i. Otherwise, let $\mu_i := \text{MMS}_{v_i}^d(\mathcal{M})$. For any good $g \in P_j^{(i)}$, $\widehat{v}_i(g) = v_i(g)/v_i(P_j^{(i)}) \leq v_i(g)/\mu_i$. Hence, $v_i(g) \geq \widehat{v}_i(g)\mu_i$. Therefore, by additivity of v, $v_i(A_i) \geq \widehat{v}_i(A_i)\mu_i$. □

Lemma 2.9 implies that `normalize`$(n, .)$ is α-MMS-preserving (when agents have additive valuations), since if A is an α-MMS allocation for the n-normalized instance $(\widehat{\mathcal{N}}, \mathcal{M}, \widehat{\mathcal{V}})$, then A is also an α-MMS allocation for the original instance $(\mathcal{N}, \mathcal{M}, \mathcal{V})$.

Definition 2.10 (Ordered instance)**.** *An instance $\mathcal{I} = (\mathcal{N}, \mathcal{M}, \mathcal{V})$ with additive valuations is* ordered *if there exists an ordering $[g_1, \dots, g_m]$ of the goods such that for all agents i, $v_i(g_1) \geq \ldots \geq v_i(g_m)$.*

We will now see how to reduce the problem of finding an α-MMS allocation or a 1-out-of-d MMS allocation to the special case of ordered instances. In fact, Barman and Krishnamurthy [13] proved that no matter what share-based fairness notion is considered, it is without loss of generality to assume the instance is ordered.

Definition 2.11. *For the fair division instance $\mathcal{I} := (\mathcal{N}, \mathcal{M}, \mathcal{V})$,* `order`$(\mathcal{I})$ *is defined as the instance $(\mathcal{N}, [|\mathcal{M}|], \widehat{\mathcal{V}})$, where for each $i \in \mathcal{N}$ and $j \in [|\mathcal{M}|]$, $\widehat{v}_i(j)$ is the j^{th} largest number in the multiset $\{v_i(g) \mid g \in \mathcal{M}\}$.*

Theorem 2.12 ([13])**.** *Given a maximin fair division instance $\mathcal{I} = ([n], \mathcal{M}, \mathcal{V})$ and scalars $\{\alpha_i \in \mathbb{R}\}_{i=1}^n$, let $([n], [|\mathcal{M}|], \widehat{\mathcal{V}})$ be the ordered instance of $\mathcal{I}$. If there exists an* *allocation* $A' = (A'_1, \ldots, A'_n)$ *that satisfies* $\hat{v}_i(A') \geq \alpha_i$*, for all* $i \in [n]$*, then there exists an allocation* $A = (A_1, \ldots, A_n)$ *in which* $v_i(A_i) \geq \alpha_i$*, for all* $i \in [n]$*. Furthermore, given* $\mathcal{I}$ *and* A'*,* A *can be computed in polynomial time.*

The proof is based on ideas by Bouveret and Lemaître [17]).

Note that ordering the instance does not change the MMS_i^d for any agent i and any integer d. Hence Corollaries 2.13 and 2.14 follow from Theorem 2.12.

Corollary 2.13 (of Theorem 2.12)**.** *For instances with additive valuations and any* $\alpha > 0$*,* `order(.)` *is* α-MMS*-preserving.*

Corollary 2.14 (of Theorem 2.12)**.** *Let* $\mathcal{I}'$ *be the ordered instance of an instance* $\mathcal{I}$ *with additive valuations. For all integers* $d > 0$*, if a* 1*-out-of-d MMS allocation exists for* $\mathcal{I}'$*, then a* 1*-out-of-d MMS allocation exists for* $\mathcal{I}$*.*

2.2 Envy-Based Notions

In envy-based class of fairness notions, agents determine the fairness through comparing their bundle with other agents' bundles.

Definition 2.15 (Envy)**.** *Upon receiving bundle* A*, we say that agent* i envies *a bundle* B*, if* $v_i(A) < v_i(B)$*. Given an allocation* X*, agent* i envies *agent* j*, if* $v_i(X_i) < v_i(X_j)$*.*

Definition 2.16 (Envy-Freeness)**.** *An allocation* X *is envy-free or* EF*, if no agent envies any other agent. In other words, for all* $i, j \in \mathcal{N}$*, we have* $v_i(X_i) \geq v_i(X_j)$*.*

Definition 2.17 (Envy-Graph)**.** *Given an allocation* X*, we define the* envy-graph *of* X *as a directed graph* $G_X = (V, E)$ *where* V *is a set of* n *nodes corresponding to agents, and there exists an edge from (the node corresponding to) agent* i *to (the node corresponding to) agent* j*, if and only if agent* i *envies agent* j*, i.e.,* $v_i(X_j) > v_i(X_i)$*.*

Similar to proportionality, in fair division of indivisible goods, envy-freeness is not a feasible guarantee. Hence, relaxations of envy-freeness has been introduced.

Definition 2.18 (Strong Envy)**.** *Upon receiving bundle* A*, we say that agent* i strongly envies *a bundle* B*, if there exists an item* $g \in B$ *such that* $v_i(A) < v_i(B \setminus g)$*. Given an allocation* X*, agent* i strongly envies *agent* j *if there exists an item* $g \in X_j$ *such that* $v_i(X_i) < v_i(X_j \setminus g)$*.*

Definition 2.19 (Envy-Freeness up to any Item (EFX) [25])**.** *An allocation* X *is* EFX *if no agent strongly envies any other agent. In other words, for all* $i, j \in \mathcal{N}$ *and all* $g \in X_j$*, we have* $v_i(X_i) \geq v_i(X_j \setminus g)$*.*

The existence of EFX allocations is one of the biggest open problems in fair division [61].

Definition 2.20 (Approximate EFX)**.** *For any* $\alpha \geq 0$*, an allocation* X *is* α-EFX*, if for all* $i, j \in \mathcal{N}$ *and all* $g \in X_j$*, we have* $v_i(X_i) \geq \alpha \cdot v_i(X_j \setminus g)$*.*

Therefore, a 1-EFX allocation is an EFX allocation.

A weaker relaxation of envy-freeness is envy-freeness up to one item.

Definition 2.21 (Envy-freeness up to One Item (EF1) [20])**.** *An allocation X is* EF1*, if for all $i, j \in \mathcal{N}$, we either have $v_i(X_i) \geq v_i(X_j)$, or there exists $g \in X_j$ such that $v_i(X_i) \geq v_i(X_j \setminus g)$.*

Similarly, for any $\alpha \geq 0$, an allocation X is α-EF1, if for all $i, j \in \mathcal{N}$, we either have $v_i(X_i) \geq \alpha \cdot v_i(X_j)$ or there exists $g \in X_j$ such that $v_i(X_i) \geq \alpha \cdot v_i(X_j \setminus g)$.

We now define the concept of *most envious agent* for a bundle.

Definition 2.22 (Most Envious Agent)**.** *Given a bundle B and a (partial) allocation X, an agent $i \in \mathcal{N}$ is a* most envious agent *of bundle B, if there exists a proper subset $B' \subsetneq B$ such that $v_i(B') > v_i(X_i)$ and no other agent $j \in \mathcal{N}$ such that $j \neq i$ strongly envies B'.*

Observation 2.23. *Given a fair division instance with additive valuations, consider a bundle B and a (partial) allocation X. If there exists an agent i who strongly envies B, then there exists an agent who is a most envious agent of B, and she can be identified in polynomial time.*

Proof. For all agents j who strongly envy B, let $B_j \subset B$ be an inclusion-wise minimal subset such that $v_j(X_j) < v_j(B)$. Let j^* be such that $|B_{j^*}|$ is minimum. Then no agent j strongly envies B_{j^*} and thus j^* is a most envious agent of B. The sets B_j's can be computed in polynomial time for each agent j by greedily removing goods from B as long as the value of the remaining set exceeds $v_j(X_j)$. And therefore, agent j^* can be identified in polynomial time as well. □

Definition 2.24 (EFX-Feasibility)**.** *Given a partition $X = (X_1, X_2, \ldots, X_n)$ of $\mathcal{M}$, a bundle X_k is EFX-feasible to agent i if and only if $v_i(X_k) \geq max_{j\in[n]} max_{g\in X_j} v_i(X_j) \setminus g$. Therefore, an allocation $X = (X_1, X_2, \ldots, X_n)$ is EFX, if for each agent i, X_i is EFX-feasible .*

Recently, Caragiannis et al. [22] introduced a promising new notion of fairness – *epistemic* EFX – by relaxing EFX, that we define next. They proved *epistemic* EFX allocations among an arbitrary number of agents with additive valuations can be computed in polynomial time.

Definition 2.25. *For any integer k, agent $i \in [n]$ and subset of items $S \subseteq \mathcal{M}$, we say that a bundle $A \subseteq S$ is "k-*epistemic-EFX*" for i with respect to S, if there exists a partitioning of $S \setminus A$ into $k-1$ bundles $C_1, C_2, \ldots, C_{k-1}$, such that for all $j \in [k-1]$, upon receiving A, i would not strongly envy C_j. We call $C = \{C_1, C_2, \ldots, C_{k-1}\}$ a "k-*certificate*" of A for i under S. Also we define*

$$\mathrm{EEFX}_i^k(S) := \{A \subseteq S \mid A \text{ is "}k\text{-epistemic-EFX"} \text{ for agent } i \text{ with respect to } S\}.$$

Definition 2.26 (EEFX)**.** *For a fair division instance, an allocation $X = (X_1, X_2, \ldots, X_n)$ is said to be* epistemic EFX *or* EEFX*, if for all agents i, $X_i \in \mathrm{EEFX}_i^n(\mathcal{M})$.*

Note that the set of EFX and EEFX allocations coincide for the case of two agents.

Chaudhury et al. [27] introduced the notion of non-degenerate instances where no agent values two distinct bundles the same. They showed that to prove the existence of EFX allocations in the additive setting, it suffices to show the existence of EFX allocations for all non-degenerate instances (with additive valuations). We adapt their approach and show that the same claim holds, even when agents have general monotone valuations.

Non-Degenerate Instances [27]. We call an instance $\mathcal{I} = \langle [n], \mathcal{M}, \mathcal{V} \rangle$ non-degenerate if and only if no agent values two different sets equally, i.e., $\forall i \in [n]$ we have $v_i(S) \neq v_i(T)$ for all $S \neq T$. We extend the technique in [27] and show that it suffices to deal with non-degenerate instances when there are n agents with general valuation functions, i.e., if there exists an EFX allocation in all non-degenerate instances, then there exists an EFX allocation in all instances.

Let $\mathcal{M} = \{g_1, g_2, \ldots, g_m\}$. We perturb any instance $\mathcal{I}$ to $\mathcal{I}(\varepsilon) = \langle [n], \mathcal{M}, \mathcal{V}(\varepsilon) \rangle$, where for every $v_i \in \mathcal{V}$ we define $v'_i \in \mathcal{V}(\varepsilon)$, as

$$v'_i(S) = v_i(S) + \varepsilon \cdot \sum_{g_j \in S} 2^j \qquad \forall S \subseteq \mathcal{M}$$

Lemma 2.27. *Let $\delta = \min_{i \in [n]} \min_{S,T:\, v_i(S) \neq v_i(T)} |v_i(S) - v_i(T)|$ and let $\varepsilon > 0$ be such that $\varepsilon \cdot 2^{m+1} < \delta$. Then*

(1) For any agent i and $S, T \subseteq \mathcal{M}$ such that $v_i(S) > v_i(T)$, we have $v'_i(S) > v'_i(T)$.

(2) $\mathcal{I}(\varepsilon)$ is a non-degenerate instance. Furthermore, if $X = (X_1, \ldots, X_n)$ is an EFX allocation for $\mathcal{I}(\varepsilon)$, then X is also an EFX allocation for $\mathcal{I}$.

Proof. For the first statement of the lemma, observe that

$$\begin{aligned} v'_i(S) - v'_i(T) &= v_i(S) - v_i(T) + \varepsilon\Big(\sum_{g_j \in S \setminus T} 2^j - \sum_{g_j \in T \setminus S} 2^j\Big) \\ &\geq \delta - \varepsilon \sum_{g_j \in T \setminus S} 2^j \\ &\geq \delta - \varepsilon \cdot (2^{m+1} - 1) \\ &> 0 \ . \end{aligned}$$

For the second statement of the lemma, consider any two sets $S, T \subseteq \mathcal{M}$ such that $S \neq T$. Now, for any $i \in [n]$, if $v_i(S) \neq v_i(T)$, we have $v'_i(S) \neq v'_i(T)$ by the first statement of the lemma. If $v_i(S) = v_i(T)$, we have $v'_i(S) - v'_i(T) = \varepsilon(\sum_{g_j \in S \setminus T} 2^j - \sum_{g_j \in T \setminus S} 2^j) \neq 0$ (as $S \neq T$). Therefore, $\mathcal{I}(\varepsilon)$ is non-degenerate.

For the final claim, let us assume that X is an EFX allocation in $\mathcal{I}(\varepsilon)$ and not an EFX allocation in $\mathcal{I}$. Then there exist i, j, and $g \in X_j$ such that $v_i(X_j \setminus g) > v_i(X_i)$. In that case, we have $v'_i(X_j \setminus g) > v'_i(X_i)$ by the first statement of the lemma, implying that X is not an EFX allocation in $\mathcal{I}(\varepsilon)$ as well, which is a contradiction. □

MMS-feasible Valuations. We introduce a new class of valuation functions called MMS-feasible valuations, which are natural extensions of additive valuations in a fair division setting.

Definition 2.28. *A valuation function $v : 2^{\mathcal{M}} \to \mathbb{R}_{\geq 0}$ is MMS-feasible, if for every subset of goods $S \subseteq \mathcal{M}$ and every partitions $A = (A_1, A_2)$ and $B = (B_1, B_2)$ of S, we have*

$$\max(v(B_1), v(B_2)) \geq \min(v(A_1), v(A_2)).$$

Informally, these are the valuations under which an agent always has a bundle in any 2-partition of any subset of the goods that she values at least as much as her MMS value, i.e., given an agent i with an MMS-feasible valuation $v(\cdot)$, in any 2-partition of $S \subseteq \mathcal{M}$, say $B = (B_1, B_2)$, we have $max(v(B_1), v(B_2)) \geq MMS_i^2(S)$, where $MMS_i^2(S)$ is the MMS value of agent i on the set S when there are 2 agents. Also, note that if there are two agents and one of the agents has an MMS-feasible valuation function, then irrespective of the valuation function of the other agent, MMS allocations always exist: Consider an instance where agent 1 has an MMS-feasible valuation function and agent 2 has a general monotone valuation function. Consider agent 2's MMS partition of the good set $A = (A_1, A_2)$. Let agent 1 pick her favourite bundle from A. Basically, agent 2 "cuts" the resource into two parts maximizing her utility assuming that she receives the bundle with minimum value, and agent 1 "chooses" her favourite part. Then, agent 1 has a bundle that she values at least as much as her MMS value (as she has an MMS-feasible valuation function), and agent 2 has a bundle that she values at least as much as her MMS value as A is an MMS partition according to agent 2. We believe that in the context of fair division, MMS-feasible form a natural class of valuation functions since they allow the basic well-known and broadly-used "cut and choose" protocol to guarantee both agents their MMS value. We note that the other popular valuation classes like submodular or subadditive do not capture this property.

Furthermore, MMS-feasible valuations strictly generalize the *cancelable valuation functions* introduced by Berger et al. [15]. A valuation function $v : 2^{\mathcal{M}} \to \mathbb{R}_{\geq 0}$ is cancelable if for every $S, T \subset M$ and $g \in M \setminus (S \cup T)$, we have $v(S \cup g) > v(T \cup g) \Rightarrow v(S) > v(T)$. Cancelable valuations include *budget-additive* ($v(S) = min(\sum_{s \in S} v(s), c)$), *unit demand* ($v(S) = max_{s \in S} v(s)$), and *multiplicative* ($v(S) = \prod_{s \in S} v(s)$) valuations [15].

Lemma 2.29. *Every cancelable function is MMS-feasible.*

We first make an observation about cancelable valuation functions.

Observation 2.30. *If v is a cancelable valuation, then for every $S, T \subset \mathcal{M}$ and $Z \subseteq \mathcal{M} \setminus (S \cup T)$, we have $v(S \cup Z) > v(T \cup Z) \Rightarrow v(S) > v(T)$.*

Proof (Lemma 2.29). Let v be a cancelable function. For a subset of goods $S \subseteq \mathcal{M}$, consider any two partitions $A = (A_1, A_2)$ and $B = (B_1, B_2)$ of S. Without loss of generality assume $v(A_1 \cap B_1) \leq v(A_2 \cap B_2)$. Since $(A_1 \cap B_2)$ is disjoint from $(A_1 \cap B_1) \cup (A_2 \cap B_2)$, by the contrapositive of Observation 2.30 applied to cancelable valuation v, we have $v((A_1 \cap B_1) \cup (A_1 \cap B_2)) \leq v((A_2 \cap B_2) \cup (A_1 \cap B_2))$. Therefore,

$$\begin{aligned}
\min(v(A_1), v(A_2)) &\leq v(A_1) \\
&= v((A_1 \cap B_1) \cup (A_1 \cap B_2)) \\
&\leq v((A_2 \cap B_2) \cup (A_1 \cap B_2)) \\
&= v(B_2) \\
&\leq \max(v(B_1), v(B_2)),
\end{aligned}$$

S	$\{g_1\}$	$\{g_2\}$	$\{g_3\}$	$\{g_1, g_2\}$	$\{g_1, g_3\}$	$\{g_2, g_3\}$	$\{g_1, g_2, g_3\}$
v	1	2	3	10	4	5	13

Table 2.1: valuation function v is MMS-feasible but not cancelable.

which proves the claim. □

To prove that MMS-feasible functions strictly generalize cancelable functions, we present an example of a valuation function that is MMS-feasible but not cancelable.

Example 2.31. *Let $\mathcal{M} = \{g_1, g_2, g_3\}$. The value of $v(S)$ is given in Table 2.1 for all $S \subseteq \mathcal{M}$. First note that $v(g_1 \cup g_2) > v(g_3 \cup g_2)$ but $v(g_1) < v(g_3)$. Therefore, v is not-cancelable. Now, we prove that v is MMS-feasible. Let $S \subseteq M$ and $A = (A_1, A_2)$, $B = (B_1, B_2)$ be two partitions of S. Without loss of generality, assume $|A_1| \leq |A_2|$. If $A_1 = \emptyset$, $\min(v(A_1), v(A_2)) = 0 \leq \max(v(B_1), v(B_2))$. Hence, we assume $|A_1| \geq 1$ and therefore, we have $|S| \geq 2$. Moreover, if $A = B$, then $\max(v(B_1), v(B_2)) = \max(v(A_1), v(A_2)) \geq \min(v(A_1), v(A_2))$. Thus, we also assume $A \neq B$. If $S = \{g, g'\}$, the only two different possible partitioning of S is $A = (\{g\}, \{g'\})$ and $B = (\emptyset, \{g, g'\})$. For all $g, g' \in \mathcal{M}$, $v(\{g, g'\}) > \max(v(g), v(g'))$. Therefore, $\max(v(B_1), v(B_2)) \geq \min(v(A_1), v(A_2))$. If $S = \{g_1, g_2, g_3\}$, then $|A_1| = 1$ and therefore, $\min(v(A_1), v(A_2)) \leq v(A_1) \leq \max_{g \in \mathcal{M}}(v(g)) = 3$. Without loss of generality, let $g_3 \in B_1$. For all $T \subseteq \mathcal{M}$ such that $g_3 \in T$, we have $v(T) \geq 3$. Thus, $\max(v(B_1), v(B_2)) \geq v(B_1) \geq 3 \geq \min(v(A_1), v(A_2))$.*

The next lemma follows from Lemma 2.29 and Example 2.31.

Lemma 2.32. *The class of MMS-feasible valuation functions is a strict superclass of cancelable valuation functions.*

PART I

Share-Based Fairness Notions

CHAPTER 3

Approximate MMS for Additive Valuations

In this chapter, we study approximate MMS allocations for instances with additive valuations. After Ghodsi et al. [44] proved the existence of 3/4-MMS allocations and gave a PTAS to compute one, Garg and Taki [42] gave a simple algorithm with complicated analysis proving the existence of $(\frac{3}{4} + \frac{1}{12n})$-MMS allocations and also computing a 3/4-MMS allocation in polynomial time.

In Section 3.3, we give a simple algorithm (with a simple analysis) that computes 3/4-MMS allocations. Then in Section 3.4, we show the existence of $(\frac{3}{4} + \frac{3}{3836})$-MMS allocations.

3.1 Notation and Tools

Throughout this section, we assume all the valuation functions are additive and instead of n-normalized, we use normalized.

Lemma 3.1. *Let $([n], [m], \mathcal{V})$ be an ordered and normalized fair division instance. For all $k \in [n]$ and agent $i \in [n]$, if $v_i(k) + v_i(2n - k + 1) > 1$, then $v_i(2n - k + 1) \leq 1/3$ and $v_i(k) > 2/3$.*

Proof. It suffices to prove $v_i(2n - k + 1) \leq 1/3$ and then $v_i(k) > 2/3$ follows. Let $P = (P_1, \dots, P_n)$ be an MMS partition of agent i. For $j \in [k]$ and $j' \in [2n + 1 - k]$, $v_i(j)+v_i(j') \geq v_i(k)+v_i(2n+1-k) > 1$, since the instance is ordered. Furthermore, j and j' cannot be in the same bundle in P since the instance is normalized. In particular, no two goods from $[k]$ are in the same bundle in P. Hence, assume without loss of generality that $j \in P_j$ for all $j \in [k]$.

For all $j \in [k]$ and $j' \in [2n-k+1], j' \notin P_j$. Thus, $\{k+1, \dots, 2n-k+1\} \subseteq P_{k+1} \cup \dots \cup P_n$. By pigeonhole principle, there exists a bundle $B \in \{P_{k+1}, \dots, P_n\}$ that contains at least 3 goods g_1, g_2, g_3 in $\{k + 1, \dots, 2n - k + 1\}$. Hence,

$$v_i(2n - k + 1) \leq \min_{g \in \{g_1,g_2,g_3\}} v_i(g) \leq \frac{1}{3} \sum_{g \in \{g_1,g_2,g_3\}} v_i(g) \leq \frac{v_i(B)}{3} = \frac{1}{3}.$$

□

3.1.1 Reduction rules

We use a technique called *valid reduction*, that helps us reduce a fair division instance to a smaller instance. This technique has been implicitly used in [6, 17, 41, 44, 53, 54] and explicitly used in [42].

Given any instance $\mathcal{I}$, a reduction rule $R(\mathcal{I})$ is a procedure that allocates a subset $S \subseteq \mathcal{M}$ of goods to an agent i and outputs the instance $\widehat{\mathcal{I}} = (\mathcal{N} \setminus \{i\}, \mathcal{M} \setminus S, \widehat{\mathcal{V}})$ where for all $j \in \mathcal{N} \setminus \{i\}$, $\widehat{v}_j = v_j$. Thus, we abuse the notation and write $\widehat{\mathcal{I}} = (\mathcal{N} \setminus \{i\}, \mathcal{M} \setminus S, \mathcal{V})$.

Definition 3.2 (Valid reductions)**.** *Let R be a reduction rule and $R(\mathcal{I}) = (\widehat{\mathcal{N}}, \widehat{\mathcal{M}}, \mathcal{V})$ such that $\{i\} = \mathcal{N} \setminus \widehat{\mathcal{N}}$ and $S = \mathcal{M} \setminus \widehat{\mathcal{M}}$. Then R is a* valid α-reduction *if*

(1) $v_i(S) \geq \alpha \cdot \text{MMS}_i^n(\mathcal{M})$, and

(2) for all $j \in \widehat{\mathcal{N}}$, $\text{MMS}_j^{n-1}(\widehat{\mathcal{M}}) \geq \text{MMS}_j^n(\mathcal{M})$.

Furthermore, a reduction rule R is a valid reduction for agent $j \in \widehat{\mathcal{N}}$, *if* $\text{MMS}_j^{n-1}(\widehat{\mathcal{M}}) \geq \text{MMS}_j^n(\mathcal{M})$ *where $\widehat{\mathcal{N}}$ and $\widehat{\mathcal{M}}$ are the set of remaining agents and remaining goods respectively after the reduction.*

Note that valid α-reductions are α-MMS-preserving, i.e., if A is an α-MMS allocation of an instance obtained by performing a valid reduction, then we can get an α-MMS allocation of the original instance by giving goods S to agent i and allocating the remaining goods as per A. A valid α-reduction, therefore, helps us reduce the problem of computing an α-MMS allocation to a smaller instance.

Lemma 3.3. *Given an instance $\mathcal{I} = (\mathcal{N}, \mathcal{M}, \mathcal{V})$, let $S \subseteq \mathcal{M}$ be such that $v_i(S) \leq \text{MMS}_i$ and $|S| \leq 2$. Then allocating S to an arbitrary agent $j \neq i$, is a valid reduction for agent i.*

Proof. Let $P = (P_1, P_2, \ldots, P_n)$ be an MMS partition of $\mathcal{M}$ for agent i. Let $g_1, g_2 \in S$. In case $|S| = 1$, $g_1 = g_2$. Without loss of generality, we assume $g_1 \in P_1$. If $g_2 \in P_1$, then $(P_2, \ldots, P_n)$ is a partition of a subset of $\mathcal{M} \setminus S$ into $n-1$ bundles with minimum value at least $\text{MMS}_i^n(\mathcal{M})$. Therefore, $\text{MMS}_i^{n-1}(\mathcal{M} \setminus S) \geq \text{MMS}_i^n(\mathcal{M})$. In case $g_2 \notin P_1$, without loss of generality, let us assume $g_2 \in P_2$. Then $v_i(P_1 \cup P_2 \setminus S) = v_i(P_1) + v_i(P_2) - v_i(S) \geq \text{MMS}_i^n$. Therefore, $(P_1 \cup P_2 \setminus S, P_3, \ldots, P_n)$ is a partition of $\mathcal{M} \setminus S$ into $n-1$ bundles with minimum value at least $\text{MMS}_i^n(\mathcal{M})$. Hence also in this case, $\text{MMS}_i^{n-1}(\mathcal{M} \setminus S) \geq \text{MMS}_i^n(\mathcal{M})$. Thus, allocating S to an arbitrary agent $j \neq i$, is a valid reduction for agent i. ☐

Now we define four reduction rules that we use in our algorithm.

Definition 3.4. *For an ordered instance $\mathcal{I} = ([n], \mathcal{M}, \mathcal{V})$ with $\text{MMS}_i = 1$ for all agents i and $\alpha > 0$, reduction rules R_1^α, R_2^α, R_3^α and R_4^α are defined as follows.*

- *$R_1^\alpha(\mathcal{I})$: If $v_i(1) \geq \alpha$ for some $i \in \mathcal{N}$, allocate $\{1\}$ to agent i and remove i from $\mathcal{N}$.*
- *$R_2^\alpha(\mathcal{I})$: If $v_i(\{2n-1, 2n, 2n+1\}) \geq \alpha$ for some $i \in \mathcal{N}$, allocate $\{2n-1, 2n, 2n+1\}$ to agent i and remove i from $\mathcal{N}$.*
- *$R_3^\alpha(\mathcal{I})$: If $v_i(\{3n-2, 3n-1, 3n, 3n+1\}) \geq \alpha$ for some $i \in \mathcal{N}$, allocate $\{3n-2, 3n-1, 3n, 3n+1\}$ to agent i and remove i from $\mathcal{N}$.*
- *$R_4^\alpha(\mathcal{I})$: If $v_i(\{1, 2n+1\}) \geq \alpha$ for some $i \in \mathcal{N}$, allocate $\{1, 2n+1\}$ to agent i and remove i from $\mathcal{N}$.*

We note that R_1^α, R_2^α, R_4^α in addition to one more rule of allocating $\{n, n+1\}$ to an agent is used in [42]. Our algorithm does not use the rule of allocating $\{n, n+1\}$. Moreover, R_3^α (allocating $\{3n-2, 3n-1, 3n, 3n+1\}$) is used in our work and not elsewhere.

Definition 3.5 (α-irreducible)**.** *For $i \in [4]$ and $\alpha > 0$, we call an instance R_i^α-irreducible, if R_i^α is not applicable. We call an instance $\mathcal{I}$ α-irreducible if none of the rules R_1^α, R_2^α, R_3^α or R_4^α is applicable.*

Lemma 3.6. *Given any $\alpha \geq 0$ and an ordered instance $\mathcal{I}$, R_1^α, R_2^α, and R_3^α are valid reductions for all the remaining agents.*

Proof. For a remaining agent i, let $P = (P_1, \ldots, P_n)$ be an MMS partition of $\mathcal{M}$ for i. It suffices to prove that after each of these reduction rules, there exists a partition of the remaining goods for each remaining agent into $n-1$ bundles with a minimum value of $\text{MMS}_i^n(\mathcal{M})$ for agent i.

- R_1^α: Let $1 \in P_k$. Then removing P_k from P results in a partition of a subset of $\mathcal{M} \setminus \{1\}$ into $n-1$ bundles of value at least $\text{MMS}_i^n(\mathcal{M})$ for agent i.
- R_2^α: By the pigeonhole principle, there exists k such that $|P_k \cap \{1, 2, \ldots, 2n+1\}| \geq 3$. Let $g_1, g_2, g_3 \in P_k \cap \{1, 2, \ldots, 2n+1\}$ and $g_1 < g_2 < g_3$. Replace g_1 with $2n-1$, g_2 with $2n$, and g_3 with $2n+1$ and remove P_k from P. Note that the value of the remaining bundles can only increase. Thus, the result is a partition of a subset of $\mathcal{M} \setminus \{2n-1, 2n, 2n+1\}$ into $n-1$ bundles with a minimum $\text{MMS}_i^n(\mathcal{M})$ for agent i.
- R_3^α: The proof is very similar to R_2^α case. By the pigeonhole principle, there exists k such that $|P_k \cap \{1, 2, \ldots, 3n+1\}| \geq 4$. Let $g_1, g_2, g_3, g_4 \in P_k \cap \{1, 2, \ldots, 3n+1\}$ and $g_1 < g_2 < g_3 < g_4$. Replace g_1 with $3n-2$, g_2 with $3n-1$, g_3 with $3n$, and g_4 with $3n+1$ and remove P_k from P. Note that the value of the remaining bundles can only increase. Thus, the result is a partition of a subset of $\mathcal{M} \setminus \{3n-2, 3n-1, 3n, 3n+1\}$ into $n-1$ bundles with a minimum value of $\text{MMS}_i^n(\mathcal{M})$ for agent i.

□

Proposition 3.7. *If $\mathcal{I} = ([n], \mathcal{M}, \mathcal{V})$ is ordered and for a given $\alpha \geq 0$ and all $j \in [3]$, $\mathcal{I}$ is R_j^α-irreducible, then*

(1) for all $k \geq 1$, $v_i(k) < \alpha$, and

(2) for all $k > 2n$, $v_i(k) < \alpha/3$, and

(3) for all $k > 3n$, $v_i(k) < \alpha/4$.

Proof. We prove each case separately.

(1) Since R_1^α is not applicable, $v_i(k) \leq v_i(1) < \alpha \cdot \text{MMS}_i$ for all agents i and all $k \geq 1$.

(2) Since R_2^α is not applicable, $3v_i(k) \leq 3v_i(2n+1) \leq v_i(2n-1) + v_i(2n) + v_i(2n+1) < \alpha \cdot \text{MMS}_i$ for all agents i and all $k > 2n$. Therefore, $v_i(k) < \alpha/3 \cdot \text{MMS}_i$.

(3) Similar to the former case, since R_3^α is not applicable, $4v_i(k) \leq 4v_i(3n+1) \leq v_i(3n-2) + v_i(3n-1) + v_i(3n) + v_i(3n+1) < \alpha \cdot \text{MMS}_i$ for all agents i and all $k > 3n$. Therefore, $v_i(k) < \alpha/4 \cdot \text{MMS}_i$.

□

Input: Instance $(\mathcal{N}, \mathcal{M}, \mathcal{V})$.
Output: Reduced instance $\mathcal{I}$.

$\mathcal{I} \leftarrow \texttt{order}(\mathcal{N}, \mathcal{M}, \mathcal{V})$
for $i \in \mathcal{N}$ **do**
  $v_i(g) \leftarrow v_i(g)/\text{MMS}_i, \forall g \in [m]$
while R_1^α or R_2^α or R_3^α or R_4^α is applicable **do**
  $\mathcal{I} \leftarrow R_k^\alpha(\mathcal{I})$ for smallest possible k
return $\mathcal{I}$

Algorithm 2: $\texttt{reduce}_\alpha((\mathcal{N}, \mathcal{M}, \mathcal{V}))$

Lemma 3.8 was proven in [42]. For completeness, we prove it here as well.

Lemma 3.8 (Lemma 3.1 in [42])**.** *For an ordered instance and for $\alpha \leq 3/4$, if the instance is R_1^α-irreducible and R_2^α-irreducible, then R_4^α is a valid α-reduction.*

Proof. It suffices to prove for all remaining agents i, R_4^α is a valid reduction. By Proposition 3.7, $v_i(1) < \alpha$ and $v_i(2n+1) < \alpha/3$. Hence, $v_i(\{1, 2n+1\}) < 4\alpha/3 \leq 1$. By Lemma 3.3, R_4^α is a valid reduction for i. ☐

Lemma 3.9. *If an ordered instance $(\mathcal{N}, \mathcal{M}, \mathcal{V})$ is $R_1(\alpha)$-irreducible for any $\alpha \leq 1$, then $|\mathcal{M}| \geq 2|\mathcal{N}|$.*

Proof. Assume $|\mathcal{M}| < 2|\mathcal{N}|$. Pick any agent $i \in \mathcal{N}$. Let P be an MMS partition of agent i. Then some bundle P_j contains a single good $\{g\}$. Then $v_i(g) = v_i(P_j) \geq \text{MMS}_i$. Hence, the instance is not $R_1(\alpha)$-irreducible for any $\alpha \leq 1$. This is a contradiction. Hence, $|\mathcal{M}| \geq 2|\mathcal{N}|$. ☐

We would like to convert fair division instances into α-irreducible instances. This can be done using a very simple algorithm, which we call $\texttt{reduce}_\alpha$. It takes a fair division instance as input, makes it ordered, scales the valuations such that for all $i \in \mathcal{N}, \text{MMS}_i = 1$, and then repeatedly applies the reduction rules R_1^α, R_2^α, R_3^α, and R_4^α until the instance becomes α-irreducible. The reduction rules can be applied in arbitrary order, except that R_4^α is only applied when R_1^α and R_2^α are inapplicable. See Algorithm 2.

Note that the application of reduction rules changes the number of agents and goods, which affects subsequent reduction rules. More precisely, the sets $\{1\}$, $\{2n-1, 2n, 2n+1\}$, $\{3n-2, 3n-1, 3n, 3n+1\}$, and $\{1, 2n+1\}$ (used in Definition 3.4) can change after applying a reduction rule. So, for example, it is possible that an instance is R_2^α-irreducible, but after applying R_3^α, the resulting instance is R_2^α-reducible.

Definition 3.10 (δ-ONI)**.** *We call an instance δ-ONI if it is ordered, normalized, and $(3/4+\delta)$-irreducible. When $\delta = 0$, we simply use ONI.*

3.2 Bag-filling Procedure

Throughout this section, let $\alpha = 3/4 + \delta$ for $\delta \geq 0$ and $\mathcal{I} = ([n], [m], \mathcal{V})$ be a fair division instance that is ordered, normalized, and α-irreducible (δ-ONI). Without loss of generality, assume that $v_i(1) \geq v_i(2) \geq \ldots \geq v_i(m)$ for each agent i.

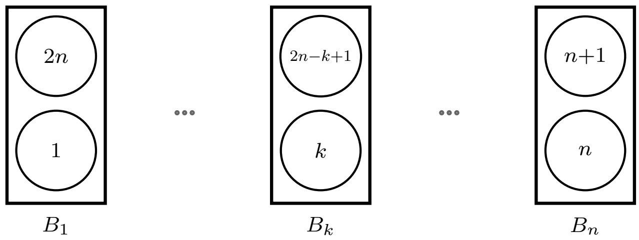


Figure 3.1: Configuration of Bags $B_1, B_2, \ldots, B_n$

Input: Ordered normalized α-irreducible instance $\mathcal{I} = ([n], [m], \mathcal{V})$ and approximation factor α.
Output: (Partial) allocation $A = (A_1, \ldots, A_n)$.

Let $(A_1, \ldots, A_n) = (\emptyset, \ldots, \emptyset)$
for $k \in [n]$ **do**
  Let $B_k = \{k, 2n + 1 - k\}$.
Let $U_G = [m] \setminus [2n]$ ▷ unassigned goods
Let $U_A = [n]$ ▷ unsatisfied agents
Let $U_B = [n]$ ▷ unassigned bags
while $U_A \neq \emptyset$ and $U_G \neq \emptyset$ **do**
  if $\exists i \in U_A$, $\exists k \in U_B$, such that $v_i(B_k) \geq \alpha$ **then**
    $A_i \leftarrow B_k$
    $U_A \leftarrow U_A \setminus \{i\}$
    $U_B \leftarrow U_B \setminus \{k\}$
  else
    Let g be an arbitrary good in U_G
    Let k be an arbitrary bag in U_B
    $B_k \leftarrow B_k \cup \{g\}$.
    $U_G \leftarrow U_G \setminus \{g\}$
return $(A_1, \ldots, A_n)$

Algorithm 3: `bagFill`$(\mathcal{I}, \alpha)$

The bag-filling procedure (`bagFill`$(\mathcal{I}, \alpha)$) was introduced by Garg, McGlaughlin, and Taki [41]. It creates n bags, where the j^{th} bag contains goods $\{j, 2n + 1 - j\}$. See Figure 3.1 for a better intuition. To create bags in this way, there must be at least $2n$ goods. This is ensured by Lemma 3.9. It then repeatedly adds a good to an arbitrary bag, and as soon as some agent i values a bag more than α, that bag is allocated to i. The algorithm terminates when all agents have been allocated a bag or all the goods are allocated. See Algorithm 3 for a more precise description. `bagFill` computes a partial allocation, i.e., some goods may remain unallocated. But that can be easily fixed by arbitrarily allocating those goods among the agents.

`bagFill`$(\mathcal{I}, \alpha)$ allocates a bag B_k to agent i only if $v_i(B_k) \geq \alpha$. Hence, to prove that `bagFill`$(\mathcal{I}, \alpha)$ returns an α-MMS allocation, it suffices to show that `bagFill` terminates successfully, i.e., all the agents receive a bag before the algorithm runs out of unallocated goods. In Section 3.3, we prove that for $\alpha = 3/4$, $\texttt{bagFill}(\mathcal{I}, \alpha)$ returns an α-MMS allocation. In this section, we prove some properties about $\texttt{bagFill}$.

For $k \in [n]$, let $B_k := \{k, 2n+1-k\}$ be the initial contents of the k^{th} bag and $\hat{B}_k$ be the k^{th} bag's contents after $\texttt{bagFill}$ terminates. We consider two groups of agents. Let $\mathcal{N}^1$ be the set of agents who value all the initial bags at most 1. Formally, $\mathcal{N}^1 := \{i \in [n] \mid \forall k \in [n], v_i(B_k) \leq 1\}$. Let $\mathcal{N}^2 := [n] \setminus \mathcal{N}^1 = \{i \in [n] \mid \exists k \in [n] : v_i(B_k) > 1\}$ be the rest of the agents. Let U_A be the set of agents that did not receive a bag when $\texttt{bagFill}$ terminated. Recall $\alpha = 3/4 + \delta$.

Lemma 3.11. *Let $i \in U_A$. For all $k \in [n]$ such that $v_i(B_k) \leq 1$, we have $v_i(\hat{B}_k) < 1 + 4\delta/3$.*

Proof. The claim trivially holds if $\hat{B}_k = B_k$. Now assume $B_k \subsetneq \hat{B}_k$. Let g be the last good added to $\hat{B}_k$. We have $v_i(\hat{B}_k \setminus g) < 3/4 + \delta$, otherwise g would not be added to $\hat{B}_k$. Also note that $g > 2n$ and hence $v_i(g) < 1/4 + \delta/3$ by Proposition 3.7. Thus, we have

$$\begin{aligned} v_i(\hat{B}_k) &= v_i(\hat{B}_k \setminus g) + v_i(g) \\ &< \left(\frac{3}{4} + \delta\right) + \left(\frac{1}{4} + \frac{\delta}{3}\right) = 1 + \frac{4\delta}{3}. \end{aligned}$$

□

Lemma 3.12. *For $\delta \leq 1/4$ and instance $\mathcal{I}$, if $\mathcal{I}$ is δ-ONI, then for all agents $i \in \mathcal{N}^2(\mathcal{I})$, $v_i(2n+1) < 1/12 + \delta$.*

Proof. By the definition of $\mathcal{N}^2$, there exist $k \in [n]$ such that $v_i(k) + v_i(2n-k+1) = v_i(B_k) > 1$. Therefore, by Lemma 3.1, $v_i(k) > 2/3$. We have,

$$\begin{aligned} v_i(2n+1) &< \frac{3}{4} + \delta - v_i(1) && (R_4^{3/4+\delta} \text{ is not applicable}) \\ &\leq \frac{3}{4} + \delta - v_i(k) && (v_i(1) \geq v_i(k)) \\ &< \frac{3}{4} + \delta - \frac{2}{3} = \frac{1}{12} + \delta, && (v_i(k) > \tfrac{2}{3}) \end{aligned}$$

which completes the proof. □

Lemma 3.13. *For $\delta \leq 1/4$ and instance $\mathcal{I}$, if $\mathcal{I}$ is δ-ONI, then for all $i \in \mathcal{N}^2$ and $k \in [n]$, $v_i(B_k) > \frac{1}{2} - 2\delta$.*

Proof. Let t be smallest such that $v_i(B_t) > 1$. By Lemma 3.1, $v_i(t) > \frac{2}{3}$. Therefore, for all $k \leq t$,

$$v_i(B_k) \geq v_i(k) \geq v_i(t) > \frac{2}{3} > \frac{1}{2} - 2\delta.$$

Note that $v_i(t) + v_i(2n-t+1) > 1$ and by Proposition 3.7, $v_i(t) < 3/4 + \delta$. Thus, $v_i(2n-t+1) > 1/4 - \delta$. For all $k > t$, we have

$$\begin{aligned} v_i(B_k) &= v_i(k) + v_i(2n-k+1) \\ &\geq 2 \cdot v_i(2n-t+1) && (v_i(k) \geq v_i(2n-k+1) \geq v_i(2n-t+1)) \\ &> \frac{1}{2} - 2\delta. \end{aligned}$$

□

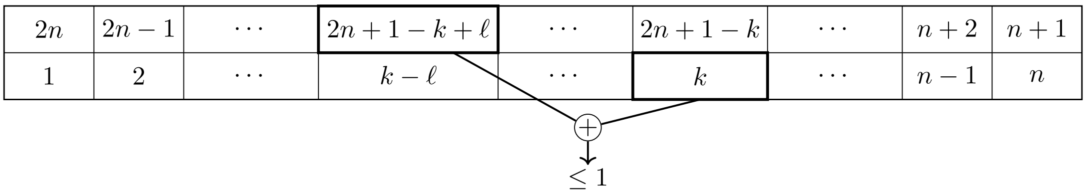


Figure 3.2: The items $[2n]$ are arranged in a table, where the k^{th} column is $B_k = \{k, 2n+1-k\}$. ℓ is the smallest *shift* such that $v_a(k) + v_a(2n+1-k+\ell) \leq 1$ for all k.

From now on, assume that $U_A \cap \mathcal{N}^2 \neq \emptyset$ and let a be a fixed agent in $U_A \cap \mathcal{N}^2$.

Let $A^+ := \{k \in [n] \mid v_a(B_k) > 1\}$, $A^- := \{k \in [n] \mid v_a(B_k) < 3/4 + \delta\}$, and $A^0 := \{k \in [n] \mid 3/4 + \delta \leq v_a(B_k) \leq 1\}$. We get upper bounds on $v_a(\hat{B}_k)$ for each of the cases $k \in A^+$, $k \in A^-$, and $k \in A^0$.

Note that $n = |A^+| + |A^-| + |A^0|$ and $|A^+| \geq 1$ since $a \in \mathcal{N}^2$.

Lemma 3.14. *For all $k \in A^-$, $v_a(\hat{B}_k) < \frac{5}{6} + 2\delta$.*

Proof. If $\hat{B}_k = B_k$, then $v_a(\hat{B}_k) < 3/4 + \delta < 5/6 + 2\delta$. Otherwise, let g be the last good added to $\hat{B}_k$. Note that $v_a(\hat{B}_k \setminus g) < 3/4 + \delta$, otherwise the algorithm would assign $\hat{B}_k \setminus g$ to agent a instead of adding g to it. We have

$$\begin{aligned} v_a(\hat{B}_k) &= v_a(\hat{B}_k \setminus g) + v_a(g) \\ &< (\frac{3}{4} + \delta) + v_a(2n+1) && (v_a(\hat{B}_k \setminus g) < \tfrac{3}{4} + \delta \text{ and } v_a(g) \leq v_a(2n+1)) \\ &< (\frac{3}{4} + \delta) + (\frac{1}{12} + \delta) = \frac{5}{6} + 2\delta. && (v_a(2n+1) < \tfrac{1}{12} + \delta \text{ by Lemma 3.12}) \end{aligned}$$

□

Let ℓ be the smallest such that for all $k \in [\ell+1, n]$, $v_a(k) + v_a(2n-k+1+\ell) \leq 1$. See Figure 3.2 for a better understanding of ℓ. Note that $\ell \geq 1$, since $a \in \mathcal{N}^2$.

Lemma 3.15. $\sum_{k \in A^+} v_a(\hat{B}_k) < |A^+| + \ell(\frac{1}{12} + \delta)$.

Proof. Let $S \in A^+$ be the set of $\min(\ell, |A^+|)$ smallest indices in A^+ and $L \in A^+$ be the set of $\min(\ell, |A^+|)$ largest indices in A^+. Since $\forall k \in A^+, \hat{B}_k = B_k$, we have

$$\sum_{k \in A^+} v_a(\hat{B}_k) = (\sum_{k \in S} v_a(k) + \sum_{k \in L} v_a(2n-k+1)) + (\sum_{k \in A^+ \setminus S} v_a(k) + \sum_{k \in A^+ \setminus L} v_a(2n-k+1)).$$

We upper bound $(\sum_{k \in S} v_a(k) + \sum_{k \in L} v_a(2n-k+1))$ and $(\sum_{k \in A^+ \setminus S} v_a(k) + \sum_{k \in A^+ \setminus L} v_a(2n-k+1))$ in Claims 3.16 and 3.17 respectively.

Claim 3.16. $\sum_{k \in S} v_a(k) + \sum_{k \in L} v_a(2n-k+1) < \ell(\frac{13}{12} + \delta)$.

Proof. Note that $v_a(k) < 3/4 + \delta$ by Proposition 3.7 and $v_a(2n-k+1) \leq 1/3$ by Lemma 3.1. Thus,

$$\sum_{k \in S} v_a(k) + \sum_{k \in L} v_a(2n-k+1) < \ell(\frac{3}{4} + \delta + \frac{1}{3}) = \ell(\frac{13}{12} + \delta).$$

■

Input: Fair division instance $\mathcal{I} = (\mathcal{N}, \mathcal{M}, \mathcal{V})$ and approximation factor α.
Output: Allocation $A = (A_1, \dots, A_n)$.

Let $\widehat{\mathcal{I}} = \texttt{order}(\texttt{normalize}(n, \texttt{reduce}_\alpha(\mathcal{I})))$
Let $\widehat{A} = \texttt{bagFill}(\widehat{\mathcal{I}}, \alpha)$.
Use $\widehat{A}$ to compute an allocation A for $\mathcal{I}$ with the same MMS approximation as $\widehat{A}$. (This can be done since `order`, $\texttt{reduce}_\alpha$, and `normalize` are α-MMS-preserving.)
return A

Algorithm 4: $\texttt{approxMMS}(\mathcal{I}, \alpha)$

Claim 3.17. *If* $\ell < |A^+|$, $\sum_{k \in A^+ \setminus S} v_a(k) + \sum_{k \in A^+ \setminus L} v_a(2n - k + 1) < |A^+| - \ell$.

Proof. Assume $A^+ = \{g_1, \dots, g_{|A^+|}\}$ and $g_1 < \dots < g_{|A^+|}$. Then, $A^+ \setminus S = \{g_{\ell+1}, \dots, g_{|A^+|}\}$ and $A^+ \setminus L = \{g_1, \dots, g_{|A^+|-\ell}\}$. The idea is to pair the goods $g_{k+\ell}$ and $2n - g_k + 1$ and prove that their value is less than 1 for agent a. Since $g_{k+\ell} \geq g_k + \ell$, $v_a(g_{k+\ell}) + v_a(2n - g_k + 1) < 1$ by the definition of ℓ. We have

$$\sum_{k \in A^+ \setminus S} v_a(k) + \sum_{k \in A^+ \setminus L} v_a(2n - k + 1) = \sum_{k \in [|A^+| - \ell]} (v_a(g_{k+\ell}) + v_a(2n - g_k + 1)) < |A^+| - \ell.$$

■

Claim 3.16 and Claim 3.17 together imply Lemma 3.15. □

Lemma 3.18. $v_a(\mathcal{M} \setminus [2n]) > \ell(\frac{1}{4} - \delta)$.

Proof. By the definition of ℓ, there exists a $k \in \{\ell, \dots, n\}$ such that $v_a(k) + v_a(2n - k + \ell) > 1$. Therefore, for all $j \leq k$ and $t \leq 2n - k + \ell$, $v_a(j) + v_a(t) > 1$. Let $P = (P_1, \dots, P_n)$ be an MMS partition of agent a. For $j \in [k]$, let $j \in P_j$. Note that for different $j, j' \in [k]$, P_j and $P_{j'}$ are different since $v_a(j) + v_a(j') > 1 = v_a(P_j)$. Also note that for every good $g \in [2n - k + \ell]$ and $j \in [k]$, $g \notin P_j$, otherwise $v_a(P_j) > 1$. Therefore, there are at least ℓ bundles like P_j among $P_1, \dots, P_k$ such that $P_j \cap [2n] = \{j\}$. We have

$$\begin{aligned} v_a(\mathcal{M} \setminus [2n]) &\geq \sum_{j \in [k]} v_a(P_j \setminus \{j\}) \geq \sum_{j \in [\ell]} (v_a(P_j) - v_a(j)) \\ &> \sum_{j \in [\ell]} \left(1 - (\frac{3}{4} + \delta)\right) = \ell(\frac{1}{4} - \delta). \qquad (v_a(j) < \tfrac{3}{4} + \delta \text{ by Proposition 3.7}) \end{aligned}$$

□

3.3 3/4-MMS Allocations

We give an algorithm, called `approxMMS` (see Algorithm 4), that takes as inputs a fair division instance and an approximation factor α. For $\alpha = 3/4$, we prove `approxMMS` returns an α-MMS allocation. It works in three major steps:

(1) Reduce the problem of finding an α-MMS allocation to the special case where the instance is ordered, normalized, and α-irreducible.

(2) Compute an α-MMS allocation for this special case using the `bagFill` algorithm (c.f. Algorithm 3).

(3) Convert this allocation for the special case to an allocation for the original fair division instance.

We describe steps 1 and 3 in Section 3.3.1 and step 2 in Section 3.3.2.

Our algorithm `approxMMS` is almost the same as the algorithm of Garg and Taki [42]. The only difference is that, unlike them, we ensure that the output of step 1 is normalized.

First, we show how to obtain an ordered normalized 3/4-irreducible instance from any arbitrary instance such that the transformation is 3/4-MMS preserving. That is, given an 3/4-MMS allocation for the resulting ordered normalized irreducible instance, one can obtain a 3/4-MMS allocation for the original instance. In the first phase of the algorithm, we obtained an ordered normalized 3/4-irreducible (INO) instance $\widehat{\mathcal{I}}$ and in the second phase, we compute a 3/4-MMS allocation for $\widehat{\mathcal{I}}$ by running $\texttt{bagFill}(\widehat{\mathcal{I}}, 3/4)$. Let $\widehat{\mathcal{I}} = ([n], [m], \mathcal{V})$.

To prove that the algorithm's output is 3/4-MMS, it suffices to prove that we never run out of goods in the bag-filling phase or, equivalently, all agents receive a bag at some point during the algorithm. To prove this, we categorize the agents into two groups as in Section 3.2. Recall $\mathcal{N}^1 = \{i \in \mathcal{N} \mid \forall k \in [n] : v_i(B_k) \leq 1\}$ and $\mathcal{N}^2 = \mathcal{N} \setminus \mathcal{N}^1 = \{i \in \mathcal{N} \mid \exists k \in [n] : v_i(B_k) > 1\}$. We note that the sets $\mathcal{N}^1$ and $\mathcal{N}^2$ are defined based on the instance $\widehat{\mathcal{I}}$ at the beginning of phase 2, and they do not change throughout the algorithm.

Agents in $\boldsymbol{\mathcal{N}^1}$. Proving that all agents in $\mathcal{N}^1$ receive a bag is easy. Using the fact that at the beginning of Phase 2, the instance is ordered, normalized, and 3/4-irreducible, we prove $v_i(g) < 1/4$ for all $i \in \mathcal{N}$ and all $g \in \mathcal{M} \setminus [2n]$. This helps to prove that any bag which is not assigned to an agent $i \in \mathcal{N}^1$ while i was available has value at most 1 to i. Therefore, since $v_i(\mathcal{M}) = n$, running out of goods is impossible before agent i receives a bag.

Agents in $\boldsymbol{\mathcal{N}^2}$. The main bulk and difficulty of the analysis of Garg and Taki [42] is to prove that all agents in $\mathcal{N}^2$ receive a bag. By normalizing the instance, we managed to simplify this argument significantly. We prove $v_i(g) < 1/12$ for all $i \in \mathcal{N}^2$ and all $g \in \mathcal{M} \setminus [2n]$. This helps to bound the value of the bags that receive some goods in the bag-filling phase by 5/6 for all available $i \in \mathcal{N}^2$. Again, if the number of such bags is high enough, it is easy to prove that the algorithm does not run out of goods in the bag-filling phase. The difficult case is when the total value of the bags which are of value more than 1 to some agent $i \in \mathcal{N}^2$ is large. Roughly speaking, in this case, it seems that the bags which receive goods in the bag-filling phase and their values are bounded by 5/6 cannot compensate for the large value of the bags that do not require any goods in the bag-filling phase. This is where the normalized property of $\widehat{\mathcal{I}}$ simplifies the matter significantly. Intuitively, there are many goods with a high value that happened to be paired in the same bag in the bag initialization phase. Since the instance is normalized, we know that in the MMS partition of i, these goods cannot be in the same bag. This implies that many bags in the MMS partition of i have at most 1 good in common with the goods in $[2n]$. This means that the value of the remaining goods (the goods in $\mathcal{M} \setminus [2n]$) must be large since they fill the bags in the MMS partition such that the value of each bag equals 1. Hence, enough goods remain in $\mathcal{M} \setminus [2n]$ to fill the bags.

3.3.1 Obtaining an Ordered Normalized 3/4-Irreducible (ONI) Instance

Lemma 3.19. *Let $\mathcal{I}$ be a fair division instance. Let*

$$\widehat{\mathcal{I}} := \texttt{order}(\texttt{normalize}(n, \texttt{reduce}_{3/4}(\mathcal{I}))).$$

Then $\widehat{\mathcal{I}}$ is ordered, normalized, and 3/4-irreducible. Furthermore, the transformation of $\mathcal{I}$ to $\widehat{\mathcal{I}}$ is 3/4-MMS-preserving, i.e., a 3/4-MMS allocation of $\widehat{\mathcal{I}}$ can be used to obtain a 3/4-MMS allocation of $\widehat{\mathcal{I}}$.

Proof. Let $\mathcal{I}^{(1)} := \texttt{reduce}_{3/4}(\mathcal{I})$. $\mathcal{I}^{(1)}$ is 3/4-irreducible and ordered, since the application of reduction rules preserves orderedness.

Let $\mathcal{I}^{(2)} := \texttt{normalize}(n, \mathcal{I}^{(1)})$. By Lemma 2.9, `normalize` does not increase the ratio of a good's value to the MMS value. Hence, $\widehat{\mathcal{I}}$ is 3/4-irreducible. $\widehat{\mathcal{I}}$ is also normalized, since for each agent, `order` only changes the identities of the goods, but the (multi-)set of values of the goods remains the same. Hence, $\widehat{\mathcal{I}}$ is ordered, normalized, and 3/4-irreducible.

Since `order`, $\texttt{reduce}_{3/4}$, and `normalize` are 3/4-MMS-preserving operations, their composition is also 3/4-MMS-preserving. □

The order of operations is important here, since `reduce` may not preserve normalizedness, and `normalize` may not preserve orderedness.

Garg and Taki [42] transform the instance as $\texttt{reduce}_{3/4}(\texttt{order}(\mathcal{I}))$, since they do not need the input to be normalized.

3.3.2 3/4-MMS Allocation of ONI Instance

Let $([n], [m], \mathcal{V})$ be a fair division instance that is ordered, normalized, and 3/4-irreducible (ONI). Without loss of generality, assume that $v_i(1) \geq v_i(2) \geq \ldots \geq v_i(m)$ for each agent i.

We run $\texttt{bagFill}(\mathcal{I}, 3/4)$. For $k \in [n]$, let $B_k := \{k, 2n+1-k\}$ be the initial contents of the k^{th} bag and $\hat{B}_k$ be the k^{th} bag's contents after `bagFill` terminates. As defined in Section 3.2, we consider two groups of agents. Let $\mathcal{N}^1$ be the set of agents who value all the initial bags at most 1. Formally, $\mathcal{N}^1 := \{i \in [n] \mid \forall k \in [n], v_i(B_k) \leq 1\}$. Let $\mathcal{N}^2 := [n] \setminus \mathcal{N}^1 = \{i \in [n] \mid \exists k \in [n] : v_i(B_k) > 1\}$ be the rest of the agents. Let U_A be the set of agents that did not receive a bag when `bagFill` terminated.

We first show that all agents in $\mathcal{N}^1$ receive a bag, i.e., $U_A \cap \mathcal{N}^1 = \emptyset$. Then we show that $U_A \cap \mathcal{N}^2 = \emptyset$. Together, these facts establish that `bagFill` terminates successfully, and hence its output is 3/4-MMS.

Lemma 3.20. *$U_A \cap \mathcal{N}^1 = \emptyset$, i.e., every agent in $\mathcal{N}^1$ gets a bag.*

Proof. For the sake of contradiction, assume $U_A \cap \mathcal{N}^1 \neq \emptyset$. Hence, $\exists i \in U_A \cap \mathcal{N}^1$. Also, for some $j \in [n]$, the j^{th} bag is unallocated. Hence, $v_i(\hat{B}_j) < 3/4$ and

$$n = v_i(\mathcal{M}) = v_i(\hat{B}_j) + \sum_{k \in [n] \setminus \{j\}} v_i(\hat{B}_k) \qquad (\text{since } \mathcal{M} = \textstyle\bigcup_{k \in [n]} \hat{B}_k)$$

$$< \frac{3}{4} + (n-1) = n - \frac{1}{4}, \qquad (\text{by Lemma 3.11})$$

which is a contradiction. Hence, $U_A \cap \mathcal{N}^1 = \emptyset$. □

Now we prove that `bagFill` allocates a bag to all agents in $\mathcal{N}^2$, i.e., $U_A \cap \mathcal{N}^2 = \emptyset$.

Lemma 3.21. *$U_A \cap \mathcal{N}^2 = \emptyset$, i.e., every agent in $\mathcal{N}^2$ gets a bag.*

Proof. Assume for the sake of contradiction that $U_A \cap \mathcal{N}^2 \neq \emptyset$. Then, as discussed in Section 3.2, we fix an agent $a \in U_A \cap \mathcal{N}^2$ and define A^+, A^-, A^0, and ℓ.

$$\begin{aligned}
n = v_a([m]) &= \sum_{k\in[n]} v_a(\hat{B}_k) \\
&= \sum_{k\in A^-} v_a(\hat{B}_k) + \sum_{k\in A^+} v_a(\hat{B}_k) + \sum_{k\in A^0} v_a(\hat{B}_k) \\
&< \frac{5}{6}|A^-| + \left(|A^+| + \frac{\ell}{12}\right) + |A^0| && \text{(by Lemmas 3.14 and 3.15)} \\
&= n + \frac{\ell}{12} - \frac{|A^-|}{6}.
\end{aligned}$$

Hence, $|A^-| < \ell/2$.

Now we show that there are enough goods in $[m] \setminus [2n]$ to fill the bags in A^-.

$$\begin{aligned}
\frac{\ell}{4} &< v_a([m] \setminus [2n]) && \text{(by Lemma 3.18)} \\
&= \sum_{k\in A^-} (v_a(\hat{B}_k) - v_a(B_k)) && \text{(since } \hat{B}_k = B_k \subseteq [2n] \text{ for } k \in A^+ \cup A^0) \\
&< |A^-| \left(\frac{5}{6} - \frac{1}{2}\right) && \text{(by Lemmas 3.14 and 3.13)} \\
&= |A^-| \cdot \frac{1}{3} < \frac{\ell}{6}, && \text{(since } |A^-| < \ell/2)
\end{aligned}$$

which is a contradiction. □

By Lemmas 3.20 and 3.21, we get that $U_A = \emptyset$, i.e., every agent gets a bag, and hence, `bagFill`'s output is 3/4-MMS.

Theorem 3.22. *Given any instance $\mathcal{I}$ with additive valuations,* `approxMMS`$(\mathcal{I}, 3/4)$ *returns a* 3/4-MMS *allocation.*

3.4 (3/4 + 3/3836)-MMS Allocations

We give an algorithm, called `mainApproxMMS` (see Algorithm 7), that takes as inputs a fair division instance and an approximation factor α. For $\alpha \leq 3/4 + 3/3836$, we prove `mainApproxMMS` returns an α-MMS allocation.

Most algorithms for approximating MMS, especially those with a factor of at least 3/4 [42, 44] and `approxMMS`$(\mathcal{I}, 3/4)$ discussed in Section 3.3, utilize two simple tools: valid reductions and bag-filling. Although these tools are easy to use in a candidate algorithm, the novelty of these works is in the analysis, which is challenging. Like previous works, the analysis is the most difficult part of our algorithm based on these tools. Unlike previous works, we also need to use a new reduction rule and initialize bags differently, which are counter-intuitive.

There are two main obstacles to generalize approxMMS (Algorithm 4) to obtain α-MMS allocations when $\alpha > 3/4$. The first obstacle lies in the first phase of the algorithm. R_4^α is a valid α-reduction when $\alpha \leq 3/4$ and R_1^α and R_2^α are not applicable. This no longer holds when $\alpha > 3/4$. In this case, the MMS value of the agents can indeed decrease after applying R_4^α. When $\alpha = 3/4 + \mathcal{O}(1/n)$, Garg and Taki [42] managed to resolve this issue by adding some dummy goods after each iteration of R_4^α and proving that the total value of these dummy goods is negligible. Essentially, since we only need to guarantee the last agent a value of α, the idea is to divide the excess $1-\alpha$ among all agents and improve the factor. However, this can only improve the factor by $\mathcal{O}(1/n)$. If $\alpha > 3/4+\varepsilon$ for a constant $\varepsilon > 0$, the same technique does not work since the value of dummy goods cannot be reasonably bounded.

We resolve this issue in Section 3.4.1. Unlike the previous works, we allow the MMS values of the remaining agents to drop. Although the MMS values of the agents can drop, we show that they do not drop by more than a multiplicative factor of $(1-4\varepsilon)$ after an arbitrary number of applications of $R_k^{3/4+\varepsilon}$ for $k \in [4]$. Basically, while for $\alpha \leq 3/4$, one can get α-irreduciblity for free (i.e., without losing any approximation factor on MMS), for $\alpha = 3/4+\varepsilon$ and $\varepsilon > 0$, we lose an approximation factor of $(1-4\varepsilon)$.

The second obstacle is that for goods in $\mathcal{M} \setminus [2n]$, we do not get the neat bound of $v_i(g) < 1/4$ for $i \in \mathcal{N}$. Instead, we get this bound with an additive factor of $\mathcal{O}(\varepsilon)$. This even complicates the analysis for agents in $\mathcal{N}^1$, which was handled very easily in Section 3.3. Furthermore, a tight example in [11, 32] shows that this algorithm cannot do better than $3/4+\mathcal{O}(1/n)$ and all the agents are in $\mathcal{N}^1$ in this example. To overcome this hurdle, we further categorize the agents in $\mathcal{N}^1$. One group consists of the agents who have a reasonable bound on the value of good $2n+1$, and the other agents, the *problematic* ones, do not.

We break the problem into two cases depending on the number of these problematic agents. In Section 3.4.3, we consider the case when the number of problematic agents is not too large. In this case, we work with a slight modification of bagFill (Algorithm 3), and using an involved analysis, we show that it gives a $(3/4+\varepsilon)$-MMS allocation. Otherwise, we introduce a new reduction rule for the first time that allocates the two most valuable goods to an agent. In Section 3.4.4, we give another algorithm to handle the case where the number of problematic agents is too large. In this case, we first apply the reduction rules (including the new one), and then initialize the bags with three goods, unlike the previous works. Precisely, we set $C_k := \{k, 2n-k+1, 2n+k\}$ and then do bag-filling.

To summarize, the structure of the rest of this section is as follows. In Section 3.4.1, given any instance $\mathcal{I} = (\mathcal{N}, \mathcal{M}, \mathcal{V})$ and $\varepsilon \in (0, 1/4)$, for $\delta \geq 4\varepsilon/(1-4\varepsilon)$ we obtain an ordered normalized $(3/4+\delta)$-irreducible (δ-ONI) instance $\widehat{\mathcal{I}} = (\widehat{\mathcal{N}}, \widehat{\mathcal{M}}, \widehat{\mathcal{V}})$ such that $\widehat{\mathcal{N}} \subseteq \mathcal{N}$, $\widehat{\mathcal{M}} \subseteq \mathcal{M}$ and all agents in $\mathcal{N} \setminus \widehat{\mathcal{N}}$ receive a bag of value at least $(3/4+\varepsilon)\text{MMS}_i(\mathcal{I})$. Moreover, we prove from any $(3/4+\delta)$-MMS allocation for $\widehat{\mathcal{I}}$, one can obtain a $\min(3/4+\varepsilon, (3/4+\delta)(1-4\varepsilon))$-MMS allocation for $\mathcal{I}$.

In Section 3.4.2, we prove a $(3/4+\delta)$-MMS allocation exists for all δ-ONI instances for any $\delta \leq 3/956$. Therefore, we prove that for $4\varepsilon/(1-4\varepsilon) \leq \delta \leq 3/956$, a $\min(3/4+\varepsilon, (3/4+\delta)(1-4\varepsilon))$-MMS exists for all instances. Setting $\delta = 3/956$ and

$\varepsilon = \delta/(4(\delta + 1)) = 3/3836$, in Section 3.4.5 we conclude that there always exists a $(3/4 + 3/3836)$-MMS allocation.

3.4.1 Reduction to δ-ONI instances

In this section, for any $\varepsilon \in (0, 1/4)$ and $\delta \geq 4\varepsilon/(1 - 4\varepsilon)$ we show how to obtain a δ-ONI instance $\widehat{\mathcal{I}}$ from any arbitrary instance $\mathcal{I}$, such that from any α-MMS allocation for $\widehat{\mathcal{I}}$, one can obtain a $\min(3/4 + \varepsilon, (1 - 4\varepsilon)\alpha)$-MMS allocation for $\mathcal{I}$. To obtain such an allocation, first, we obtain a $(3/4 + \varepsilon)$-irreducible instance, and we prove that the MMS value of no remaining agent drops by more than a multiplicative factor of $(1 - 4\varepsilon)$. Then, we normalize and order the resulting instance, giving us a δ-ONI instance (for $\delta \geq 4\varepsilon/(1 - 4\varepsilon)$). In the rest of this section, by R_k we mean $R_k^{(3/4+\varepsilon)}$ for $k \in [4]$.

We start with reducing the instance using $\texttt{reduce}_{3/4+\varepsilon}$. I.e., we transform the instance into an ordered one using the `order` algorithm. Then, as long as one of the reduction rules R_1, R_2, R_3, or R_4 is applicable, we apply R_k for the smallest possible k. Algorithm 2 shows the pseudocode of this procedure.

In this section, we prove the following two theorems.

Theorem 3.23. *Given an instance $\mathcal{I} = (\mathcal{N}, \mathcal{M}, \mathcal{V})$ and $\varepsilon \geq 0$, let $\widehat{\mathcal{I}} = (\widehat{\mathcal{N}}, \widehat{\mathcal{M}}, \widehat{\mathcal{V}}) = \texttt{reduce}_{3/4+\varepsilon}(\mathcal{I})$. For all agents $i \in \widehat{\mathcal{N}}$, $\text{MMS}_i(\widehat{\mathcal{I}}) \geq (1 - 4\varepsilon)$.*

Theorem 3.24. *Given an instance $\mathcal{I}$ and $\varepsilon \geq 0$, let $\widehat{\mathcal{I}} = \texttt{order}(\texttt{normalize}(\texttt{reduce}_{3/4+\varepsilon}(\mathcal{I})))$. Then $\widehat{\mathcal{I}}$ is ordered, normalized and $(\frac{3}{4} + \frac{4\varepsilon}{1-4\varepsilon})$-irreducible ($\frac{4\varepsilon}{1-4\varepsilon}$-ONI). Furthermore, from any α-MMS allocation of $\widehat{\mathcal{I}}$ one can obtain a $\min(3/4 + \varepsilon, (1 - 4\varepsilon)\alpha)$-MMS allocation of $\mathcal{I}$.*

Note that once R_1 is not applicable, we have $v_i(1) < 3/4 + \varepsilon$ for all remaining agents i. Since we never increase the values, R_1 can no longer apply. So $\texttt{reduce}_{3/4+\varepsilon}(\mathcal{I})$ first applies R_1 as long as it is applicable and then applies the rest of the reduction rules. Since R_1 is a valid reduction rule for all the remaining agents i by Lemma 3.6, $\text{MMS}_i \geq 1$ after applications of R_1. So to prove Theorem 3.23 without loss of generality, we assume R_1 is not applicable on $\mathcal{I} = ([n], M, \mathcal{V})$. Let $\widehat{\mathcal{I}} = (\widehat{\mathcal{N}}, \widehat{\mathcal{M}}, \mathcal{V}) = \texttt{reduce}_{3/4+\varepsilon}(\mathcal{I})$. For the rest of this section, we fix agent $i \in \widehat{\mathcal{N}}$. Let $P = (P_1, P_2, \ldots, P_n)$ be the initial MMS partition of i (in $\mathcal{I}$). We construct a partition $Q = (Q_1, Q_2, \ldots, Q_{|\widehat{\mathcal{N}}|})$ of $\widehat{\mathcal{M}}$ such that $v_i(Q_j) \geq 1 - 4\varepsilon$ for all $j \in [|\widehat{\mathcal{N}}|]$.

Let G_2, G_3, and G_4 be the set of goods removed by applications of R_2, R_3, and R_4, respectively. Also, let $r_2 = |G_2|/3$, $r_3 = |G_3|/4$, and $r_4 = |G_4|/2$ be the number of times each rule is applied, respectively. Note that in the end, all that matters is that we construct a partition Q of $\mathcal{M} \setminus (G_2 \cup G_3 \cup G_4)$ into $n - (r_2 + r_3 + r_4)$ bundles of value at least $1 - 4\varepsilon$ for i. For this sake, it does not matter in which order the goods are removed. Therefore, without loss of generality, we assume all the goods in G_4 are removed first, and then the goods in G_2 and G_3 are removed in their original order. Note that we are not applying the reduction rules in a different order. We are removing the same goods that would be removed by applying the reduction rules in their original order. Only for the sake of our analysis, we remove these goods in a different order. For better intuition, consider the following example. Assume $\texttt{reduce}_{3/4+\varepsilon}(\mathcal{I})$ first applies R_2

removing $\{a_1, a_2, a_3\}$, then R_4 removing $\{b_1, b_2\}$, then another R_2 removing $\{c_1, c_2, c_3\}$ and then R_3 removing $\{d_1, d_2, d_3, d_4\}$. Without loss of generality we can assume that first $\{b_1, b_2\}$ is removed, then $\{a_1, a_2, a_3\}$, then $\{c_1, c_2, c_3\}$ and then $\{d_1, d_2, d_3, d_4\}$.

We know that when there are n agents, removing $\{2n-1, 2n, 2n+1\}$ (or $\{3n-2, 3n-1, 3n, 3n+1\}$) and an agent is a valid reduction for i by Lemma 3.6. With the same argument, it is not difficult to see that removing $\{g_1, g_2, g_3\}$ where $g_1 \geq 2n-1$, $g_2 \geq 2n$ and $g_3 \geq 2n+1$ (or $\{g_1, g_2, g_3, g_4\}$ where $g_1 \geq 3n-2$, $g_2 \geq 3n-1$, $g_3 \geq 3n$ and $g_4 \geq 3n+1$) and an agent is also a valid reduction for i. For completeness, we prove this in Lemma 3.25.

Lemma 3.25. *Let $\mathcal{I} = (\mathcal{N}, \mathcal{M}, \mathcal{V})$ be an ordered instance and $i \in \mathcal{N}$.*

(1) Let $g_1 \geq 2n-1$, $g_2 \geq 2n$ and $g_3 \geq 2n+1$. Then $\text{MMS}_{v_i}^{n-1}(\mathcal{M} \setminus \{g_1, g_2, g_3\}) \geq \text{MMS}_{v_i}^{n}(\mathcal{M})$.

(2) Let $g_1 \geq 3n-2$, $g_2 \geq 3n-1$, $g_3 \geq 3n$ and $g_4 \geq 3n+1$. Then $\text{MMS}_{v_i}^{n-1}(\mathcal{M} \setminus \{g_1, g_2, g_3, g_4\}) \geq \text{MMS}_{v_i}^{n}(\mathcal{M})$.

Proof. (1) By the pigeonhole principle, there exists k such that $|P_k \cap \{1, 2, \dots, 2n+1\}| \geq 3$. Let $h_1, h_2, h_3 \in P_k \cap \{1, 2, \dots, 2n+1\}$ and $h_1 < h_2 < h_3$. Replace h_1 with g_1, h_2 with g_2 and h_3 with g_3 and remove P_k from P. Note that the value of the remaining bundles can only increase. Thus, the result is a partition of a subset of $\mathcal{M} \setminus \{g_1, g_2, g_3\}$ into $n-1$ bundles with a minimum value of $\text{MMS}_i^n(\mathcal{M})$ for agent i.

(2) By the pigeonhole principle, there exists k such that $|P_k \cap \{1, 2, \dots, 3n+1\}| \geq 4$. Let $h_1, h_2, h_3, h_4 \in P_k \cap \{1, 2, \dots, 3n+1\}$ and $h_1 < h_2 < h_3 < h_4$. Replace h_1 with g_1, h_2 with g_2, h_3 with g_3 and h_4 with g_4 and remove P_k from P. Note that the value of the remaining bundles can only increase. Thus, the result is a partition of a subset of $\mathcal{M} \setminus \{g_1, g_2, g_3, g_4\}$ into $n-1$ bundles with a minimum value of $\text{MMS}_i^n(\mathcal{M})$ for agent i.

□

Observation 3.26. *Given an ordered instance $\mathcal{I} = (\mathcal{N}, \mathcal{M}, \mathcal{V})$, let $v_i(g_1) \geq \dots \geq v_i(g_m), \forall i \in \mathcal{N}$. Let $\widehat{\mathcal{I}} = (\widehat{\mathcal{N}}, \widehat{\mathcal{M}}, \mathcal{V})$ be the instance after removing an agent i and a set of goods $\{a, b\}$ from $\mathcal{I}$. Let $g \in \widehat{\mathcal{M}}$ be the j^{th} most valuable good in $\mathcal{M}$ and the j'^{th} most valuable good in $\widehat{\mathcal{M}}$. Then $j' \geq j-2$.*

Corollary 3.27 (of Observation 3.26)**.** *Given an ordered instance $\mathcal{I} = (\mathcal{N}, \mathcal{M}, \mathcal{V})$, let $\widehat{\mathcal{I}} = (\widehat{\mathcal{N}}, \widehat{\mathcal{M}}, \mathcal{V})$ be the instance after removing an agent i and a set of goods $\{a, b\}$ from $\mathcal{I}$. Let $n = |\mathcal{N}|$ and $n' = |\widehat{\mathcal{N}}| = n-1$. Let $g \in \widehat{\mathcal{M}}$ be the j^{th} most valuable good in $\mathcal{M}$ and the j'^{th} most valuable good in $\widehat{\mathcal{M}}$. Then,*

- *for any k, in particular, $k \in \{-1, 0, 1\}$, if $j \geq 2n+k$, then $j' \geq 2n'+k$, and*
- *for any k, in particular, $k \in \{-2, -1, 0, 1\}$, if $j \geq 3n+k$, then $j' \geq 3n'+k$.*

Next, assume at a step where the number of agents is n, $\{g_{2n-1}, g_{2n}, g_{2n+1}\}$ (or $\{g_{3n-2}, g_{3n-1}, g_{3n}, g_{3n+1}\}$) is removed with an application of R_2 (or R_3). Corollary 3.27 together with Lemma 3.25 imply that removing $\{g_{2n-1}, g_{2n}, g_{2n+1}\}$ (or

$\{g_{3n-2}, g_{3n-1}, g_{3n}, g_{3n+1}\}$) at a later step where the number of agents is $n' \leq n$ is also valid for agent i. Therefore, all that remains is to prove that after removing the goods in G_4 and r_4 agents, the MMS value of i remains at least $1-4\varepsilon$. That is, $\text{MMS}_i^{n-r_4}(M \setminus G_4) \geq 1-4\varepsilon$.

Lemma 3.28. *Let* $(\widehat{\mathcal{N}}, \widehat{\mathcal{M}}, \mathcal{V}) = \texttt{reduce}_{3/4+\varepsilon}(([n], \mathcal{M}, \mathcal{V}))$. *Let* r_4 *be the number of times* R_4 *is applied during* $\texttt{reduce}_{3/4+\varepsilon}(\mathcal{I})$ *and let* G_4 *be the set of removed goods by applications of* R_4. *Then for all agents* $i \in \widehat{\mathcal{N}}$, $\text{MMS}_{v_i}^{n-r_4}(\mathcal{M} \setminus G_4) \geq 1 - 4\varepsilon$.

Proof. Without loss of generality, assume all the goods in G_4 are in $P_1 \cup P_2 \cup \ldots \cup P_k$ for some $k \leq 2r_4$. Namely, we have $P_j \cap G_4 \neq \emptyset$ for all $j \in [k]$ and $(P_{k+1} \cup \ldots \cup P_n) \cap G_4 = \emptyset$. If $k \leq r_4$, then $(P_{k+1}, \ldots, P_n)$ is already a partition of a subset of $\mathcal{M} \setminus G_4$ into at least $n - r_4$ bundles. Therefore the lemma follows.

So assume $k > r_4$. In each application of R_4, two goods h and ℓ are removed. Let h be the more valuable good. We call h the heavy good and ℓ the light good of this application of R_4. By Proposition 3.7, for all heavy goods h and light goods ℓ we have, $v_i(h) < 3/4+\varepsilon$ and $v_i(\ell) < 1/4 + \varepsilon/3$. Let H be the set of all heavy goods and L be the set of all light goods removed during these reductions. Hence, $G_4 = H \cup L$, $|H| = |L| = r_4$.

We prove that we can partition $(P_1 \cup \ldots P_k) \setminus G_4$ into $k - r_4$ bundles $Q_1, \ldots, Q_{k-r_4}$, each of value at least $1 - 4\varepsilon$. Or equivalently $\text{MMS}_{v_i}^{k-r_4}\big((P_1 \cup \ldots \cup P_k) \setminus G_4\big) \geq 1 - 4\varepsilon$. Then, $(Q_1, \ldots, Q_{k-r_4}, P_{k+1}, \ldots, P_n)$ is a partition of $\mathcal{M} \setminus G_4$ into $n - r_4$ bundles, each of value at least $1 - 4\varepsilon$ and the lemma follows. It suffices to prove the following claim.

Claim 3.29. *For* $r < k \leq 2r$, *if* $|(P_1 \cup P_2 \cup \ldots \cup P_k) \cap H| \leq r$ *and* $|(P_1 \cup P_2 \cup \ldots \cup P_k) \cap L| \leq r$, *then* $\text{MMS}_{v_i}^{k-r}\big((P_1 \cup \ldots \cup P_k) \setminus G_4\big) \geq 1 - 4\varepsilon$ *for all* $0 < r < k$.

The proof of Claim 3.29 is by induction on k. For $k = 2$, we have $r = 1$ and $v_i(P_1 \cup P_2) - v_i(H \cup L) \geq 2 - (\frac{3}{4} + \varepsilon) - (\frac{1}{4} + \frac{\varepsilon}{3}) > 1 - 4\varepsilon$ and therefore, $\text{MMS}_{v_i}^1(P_1 \cup P_2 \setminus G_4) \geq 1 - 4\varepsilon$. Now assume that the statement holds for all values of $k' \leq k - 1$, and we prove it for $k > 2$. First, we prove the claim when at least one of the inequalities is strict. Assume $|(P_1 \cup P_2 \cup \ldots \cup P_k) \cap H| < r$ and $|(P_1 \cup P_2 \cup \ldots \cup P_k) \cap L| \leq r$. The proof of the other case is symmetric. If $(P_1 \cup P_2 \cup \ldots \cup P_k) \cap L \neq \emptyset$, without loss of generality assume $P_k \cap L \neq \emptyset$. Therefore, $|(P_1 \cup \ldots \cup P_{k-1}) \cap H| \leq r - 1 < k - 1$ and $|(P_1 \cup \ldots \cup P_{k-1}) \cap L| \leq r - 1 < k - 1$. We have,

$$\begin{aligned} \text{MMS}_{v_i}^{k-r}\big((P_1 \cup \ldots \cup P_k) \setminus G_4\big) &\geq \text{MMS}_{v_i}^{(k-1)-(r-1)}\big((P_1 \cup \ldots \cup P_{k-1}) \setminus G_4) \\ &\geq 1 - 4\varepsilon. \qquad \text{(by induction assumption)} \end{aligned}$$

Now assume $|(P_1 \cup P_2 \cup \ldots \cup P_k) \cap H| = r$ and $|(P_1 \cup P_2 \cup \ldots \cup P_k) \cap L| = r$.

Case 1: There exists $j \in [k]$, such that $P_j \cap H \neq \emptyset$ and $P_j \cap L \neq \emptyset$. Without loss of generality, assume $P_k \cap H \neq \emptyset$ and $P_k \cap L \neq \emptyset$. In this case $|(P_1 \cup \ldots \cup P_{k-1}) \cap H| \leq r - 1 < k - 1$ and $|(P_1 \cup \ldots \cup P_{k-1}) \cap L| \leq r - 1 < k - 1$. Therefore,

$$\begin{aligned} \text{MMS}_{v_i}^{k-r}\big((P_1 \cup \ldots \cup P_k) \setminus G_4\big) &\geq \text{MMS}_{v_i}^{(k-1)-(r-1)}\big((P_1 \cup \ldots \cup P_{k-1}) \setminus G_4\big) \\ &\geq 1 - 4\varepsilon. \qquad \text{(by induction assumption)} \end{aligned}$$

Case 2: There exist $j, \ell \in [k]$, such that $|P_j \cap H| \geq 2$ and $|P_\ell \cap L| \geq 2$. Similar to the former case, we have

$$\begin{aligned}\mathrm{MMS}^{k-r}_{v_i}\big((P_1 \cup \ldots \cup P_k) \setminus G_4\big) &\geq \mathrm{MMS}^{(k-2)-(r-2)}_{v_i}\big((P_1 \cup \ldots \cup P_{k-2}) \setminus G_4\big) \\ &\geq 1 - 4\varepsilon. \qquad \text{(by induction assumption)}\end{aligned}$$

Case 3: Neither Case 1 nor Case 2 holds. For all $j \in [k]$, we have $P_j \cap H = \emptyset$ or $P_j \cap L = \emptyset$; otherwise, we are in case 1. Let $S_1 := \{j \in [k] \mid P_j \cap L \neq \emptyset\}$ and $S_2 = [k] \setminus S_1 = \{j \in [k] \mid P_j \cap H \neq \emptyset\}$. If there exist bundles P_j and P_ℓ such that $|P_j \cap H| \geq 2$ and $|P_\ell \cap L| \geq 2$, we are in case 2. Therefore, for all $j \in S_1$, $|P_j \cap L| = 1$ or for all $j \in S_2$, $|P_j \cap H| = 1$. Hence, there are r bundles $P_1, \ldots, P_r$ such that either $|P_j \cap H| = 1$ (and $|P_j \cap L| = 0$) for all $j \in [r]$ or $|P_j \cap L| = 1$ (and $|P_j \cap H| = 0$) for all $j \in [r]$.

Case 3.1: $k > r + 1$. Assume $|P_j \cap H| = 1$ for all $j \in [r]$. (The case where $|P_j \cap L| = 1$ for all $j \in [r]$ is symmetric when $k > r + 1$.) Let $|P_k \cap L| = a$. Then $|(P_1 \cup \ldots \cup P_a \cup P_k) \cap H| = a$ and $|(P_1 \cup \ldots \cup P_a \cup P_k) \cap L| = a$. Thus by the induction assumption, we have

$$\mathrm{MMS}^{(a+1)-a}_{v_i}\big((P_1 \cup \ldots \cup P_a \cup P_k) \setminus G_4\big) \geq 1 - 4\varepsilon.$$

Moreover, $|(P_{a+1} \cup \ldots \cup P_{k-1}) \cap H| \leq r - a$ and $|(P_{a+1} \cup \ldots \cup P_{k-1}) \cap L| \leq r - a$. Thus, by the induction assumption we have

$$\mathrm{MMS}^{(k-a-1)-(r-a)}_{v_i}\big((P_{a+1} \cup \ldots \cup P_{k-1}) \setminus G_4\big) \geq 1 - 4\varepsilon.$$

So we can partition $(P_1 \cup \ldots \cup P_a \cup P_k) \setminus G_4$ into one bundle of value at least $1 - 4\varepsilon$ for i and also we can partition $(P_{a+1} \cup \ldots \cup P_{k-1}) \setminus G_4$ into $k - r - 1$ bundles of value at least $1 - 4\varepsilon$ for i. Thus, the lemma holds.

Case 3.2: $k = r + 1$. Let $B = (P_1 \cup \ldots \cup P_k) \setminus G_4$. We want to show $\mathrm{MMS}^1_{v_i}(B) \geq 1 - 4\varepsilon$. Hence it suffices to show $v_i(B) \geq 1 - 4\varepsilon$.

$$\begin{aligned}v_i(B) &\geq \sum_{j \in [k-1]} v_i\left(P_j \setminus (H \cup L)\right) \\ &= \sum_{j \in [k-1]} \left(v_i(P_j) - v_i(P_j \cap (H \cup L))\right) \\ &> (k-1)\left(1 - (\frac{3}{4} + \varepsilon)\right) \qquad \text{(since } |P_j \cap (H \cup L)| = 1,\ v_i(P_j \cap (H \cup L)) \leq \tfrac{3}{4} + \varepsilon) \\ &= (k-1)(\frac{1}{4} - \varepsilon) \geq 1 - 4\varepsilon. \qquad \text{(for } k > 4)\end{aligned}$$

It remains to prove the claim when $k = 3$ and $k = 4$. If there are two bundles P_1 and P_2 such that $|P_1 \cap L| = |P_2 \cap L| = 1$, $v_i(B) \geq v_i(P_1 \setminus L) + v_i(P_2 \setminus L) > 2\left(1 - (\frac{1}{4} + \frac{\varepsilon}{3})\right) > 1 - 4\varepsilon$.

Otherwise, for $k = 3$, there are two bundles P_1 and P_2 such that $|P_1 \cap H| = |P_2 \cap H| = 1$ and $|P_3 \cap L| = 2$. Then,

$$\begin{aligned} v_i(B) &= v_i(P_1 \setminus H) + v_i(P_2 \setminus H) + v_i(P_3 \setminus L) \\ &> 2\left(1 - (\frac{3}{4} + \varepsilon)\right) + \left(1 - 2(\frac{1}{4} + \frac{\varepsilon}{3})\right) \\ &= 1 - \frac{8\varepsilon}{3} > 1 - 4\varepsilon. \end{aligned}$$

For $k = 4$, we have $|P_1 \cap H| = |P_2 \cap H| = |P_3 \cap H| = 1$ and $|P_4 \cap L| = 3$. Then,

$$\begin{aligned} v_i(B) &= v_i(P_1 \setminus H) + v_i(P_2 \setminus H) + v_i(P_3 \setminus H) + v_i(P_4 \setminus L) \\ &> 3\left(1 - (\frac{3}{4} + \varepsilon)\right) + \left(1 - 3(\frac{1}{4} + \frac{\varepsilon}{3})\right) = 1 - 4\varepsilon. \end{aligned}$$

□

We are ready to prove Theorem 3.23 and Theorem 3.24.

Theorem 3.23. *Given an instance* $\mathcal{I} = (\mathcal{N}, \mathcal{M}, \mathcal{V})$ *and* $\varepsilon \geq 0$*, let* $\widehat{\mathcal{I}} = (\widehat{\mathcal{N}}, \widehat{\mathcal{M}}, \widehat{\mathcal{V}}) = \texttt{reduce}_{3/4+\varepsilon}(\mathcal{I})$*. For all agents* $i \in \widehat{\mathcal{N}}$*,* $\text{MMS}_i(\widehat{\mathcal{I}}) \geq (1 - 4\varepsilon)$*.*

Proof. Fix an agent $i \in \widehat{\mathcal{N}}$. Let $\mathcal{I}^{(1)}$ be the instance after all applications of R_1 and before any further reduction. By Lemma 3.6, $\text{MMS}_i(\mathcal{I}^{(1)}) \geq 1$. So without loss of generality, let us assume $\mathcal{I} = \mathcal{I}^{(1)}$. Let G_2, G_3, and G_4 be the set of goods removed by applications of R_2, R_3, and R_4, respectively. Also, let $r_2 = |G_2|/3$, $r_3 = |G_3|/4$, and $r_4 = |G_4|/2$ be the number of times each rule is applied, respectively. By Lemma 3.28, $\text{MMS}^{n-r_4}_{v_i}(\mathcal{M} \setminus G_4) \geq 1 - 4\varepsilon$. For an application of R_2 (or R_3) at step t, let $\{a_1, a_2, a_3\}$ (or $\{b_1, b_2, b_3, b_4\}$) be the set of goods that are removed. By Lemma 3.25, removing this set at a step $t' \geq t$ is still a valid reduction for i. Therefore, removing G_2 and G_3 and $r_2 + r_3$ agents does not decrease the MMS value of i. Thus, $\text{MMS}_i(\widehat{\mathcal{I}}) \geq 1 - 4\varepsilon$. □

Theorem 3.24. *Given an instance* $\mathcal{I}$ *and* $\varepsilon \geq 0$*, let* $\widehat{\mathcal{I}} = \texttt{order}(\texttt{normalize}(\texttt{reduce}_{3/4+\varepsilon}(\mathcal{I})))$*. Then* $\widehat{\mathcal{I}}$ *is ordered, normalized and* $(\frac{3}{4} + \frac{4\varepsilon}{1-4\varepsilon})$*-irreducible* ($\frac{4\varepsilon}{1-4\varepsilon}$*-ONI). Furthermore, from any* α*-MMS allocation of* $\widehat{\mathcal{I}}$ *one can obtain a* $\min(3/4 + \varepsilon, (1 - 4\varepsilon)\alpha)$*-MMS allocation of* $\mathcal{I}$*.*

Proof. In $\texttt{reduce}_{3/4+\varepsilon}$, as long as R_1 is applicable, we apply it. Once it is not applicable anymore, for all remaining agents i, $v_i(1) < 3/4 + \varepsilon$. In the rest of procedure $\texttt{reduce}_{3/4+\varepsilon}$, we do not increase the value of any good for any agent. Therefore, R_1 remains inapplicable. As long as one of the rules R_k is applicable for $k \in \{2, 3, 4\}$, we apply it. Therefore, $\texttt{reduce}_{3/4+\varepsilon}(\mathcal{I})$ is $(3/4 + \varepsilon)$-irreducible. Let $\mathcal{I}' = \texttt{reduce}(\mathcal{I}, \varepsilon)$. Since $\text{MMS}_i(\mathcal{I}') \geq 1 - 4\varepsilon$ (by Theorem 4.1), $\texttt{normalize}$ can increase the value of each good by a multiplicative factor of at most $1/(1 - 4\varepsilon)$. Therefore, after ordering the instance, none of the rules R^α_k for $k \in [4]$ would be applicable for $\alpha \geq \frac{3/4+\varepsilon}{1-4\varepsilon} = \frac{3}{4} + \frac{4\varepsilon}{1-4\varepsilon}$. Hence, $\widehat{\mathcal{I}} = \texttt{order}(\texttt{normalize}(\texttt{reduce}_{3/4+\varepsilon}(\mathcal{I})))$ is α-irreducible for $\alpha \geq \frac{3}{4} + \frac{4\varepsilon}{1-4\varepsilon}$ and it is of course ordered. Since $\texttt{order}$ does not change the multiset of the values of the goods for each agent, the instance remains normalized.

Now let us assume A is an α-MMS allocation for $\widehat{\mathcal{I}} = \texttt{order}(\texttt{normalize}(\texttt{reduce}_{3/4+\varepsilon}(\mathcal{I})))$. By Corollary 2.13, we can obtain an allocation B which is α-MMS for $\texttt{normalize}(\texttt{reduce}_{3/4+\varepsilon}(\mathcal{I}))$. Lemma 2.9 implies that B is α-MMS for $\mathcal{I}' = (\mathcal{N}', \mathcal{M}', \mathcal{V}') = \texttt{reduce}_{3/4+\varepsilon}(\mathcal{I})$. For all agents $i \in \mathcal{N}'$, $v_i'(B_i) = v_i(B_i)/\text{MMS}_i(\mathcal{I})$. Therefore,

$$\begin{aligned} v_i(B_i) &= v_i'(B_i) \cdot \text{MMS}_i(\mathcal{I}) \\ &\geq \alpha \cdot \text{MMS}_i(\mathcal{I}') \cdot \text{MMS}_i(\mathcal{I}) && (B \text{ is } \alpha\text{-MMS for } \mathcal{I}') \\ &\geq \alpha(1-4\varepsilon) \cdot \text{MMS}^n_{v_i}(\mathcal{M}). && (\text{MMS}_i(\mathcal{I}') \geq 1-4\varepsilon \text{ by Theorem 3.23}) \end{aligned}$$

Thus, B gives all the agents in $\mathcal{N}'$, $\alpha(1-4\varepsilon)$ fraction of their MMS. All agents in $\mathcal{N} \setminus \mathcal{N}'$ receive $(3/4+\varepsilon)$ fraction of their MMS value. Therefore, the final allocation is a $\min(3/4+\varepsilon, (1-4\varepsilon)\alpha)$-MMS allocation of $\mathcal{I}$. □

3.4.2 $(3/4+\delta)$-MMS Allocation for δ-ONI Instances

In this section, we prove that for $\delta \leq 3/956$ there exists a $(3/4+\delta)$-MMS allocation if the input is a δ-ONI instance.

We initialize n bags $\{B_1, \dots, B_n\}$ with the first $2n$ goods as follows:

$$B_k := \{k, 2n-k+1\} \text{ for } k \in [n]. \tag{3.1}$$

See Figure 3.1 for a better intuition. Note that by Lemma 3.9, $m \geq 2n$ and such bag-initialization is possible.

Given an instance $\mathcal{I} = ([n], [m], \mathcal{V})$ (with $m \geq 2n$), let $\mathcal{N}^1(\mathcal{I}) = \{i \in [n] \mid \forall k \in [n] : v_i(B_k) \leq 1\}$ and $\mathcal{N}^2(\mathcal{I}) = \{i \in [n] \mid \exists k \in [n] : v_i(B_k) > 1\}$.

We refer to $\mathcal{N}^1(\mathcal{I})$ and $\mathcal{N}^2(\mathcal{I})$ by $\mathcal{N}^1$ and $\mathcal{N}^2$ respectively when $\mathcal{I}$ is the initial δ-ONI instance. Recall that $\mathcal{N}^1$ and $\mathcal{N}^2$ do not change over the course of our algorithm. Let $\mathcal{N}^1_1 = \{i \in \mathcal{N}^1 \mid v_i(2n+1) \geq 1/4 - 5\delta\}$ and $\mathcal{N}^1_2 = \mathcal{N}^1 \setminus \mathcal{N}^1_1$. Depending on the number of agents in $\mathcal{N}^1_1$, we run one of the $\texttt{approxMMS1}(\mathcal{I}, \delta)$ or $\texttt{approxMMS2}(\mathcal{I}, \delta)$ shown in Algorithms 5 and 6 respectively. Roughly speaking, if the size of $\mathcal{N}^1_1$ is not too large, we run Algorithm 5 and prioritize agents in $\mathcal{N}^1_1$. Otherwise, we run Algorithm 6 giving priority to agents in $\mathcal{N}^1_2 \cup \mathcal{N}^2$. Giving priority to agents in a certain set S means that we break the tie in favour of agents in S. In other words, when the algorithm is about to allocate a bag B to an agent, if there is an agent in S who gets satisfied upon receiving B (i.e., $v_i(B) \geq 3/4 + \delta$ for some $i \in S$), then the algorithms give B to such an agent and not to someone outside S.

3.4.3 Case 1: $|\mathcal{N}^1_1| \leq n(\frac{1}{4} - \delta)/(\frac{1}{4} + \frac{\delta}{3})$

In this case we run Algorithm 5 which is similar to the bag-filling procedure discussed in Section 3.2, with the difference that we always break ties in favour of agents in $\mathcal{N}^1_1$. For $k \in [n]$, let B_k and $\hat{B}_k \supseteq B_k$ be the k^{th} bag at the beginning and end of Algorithm 5, respectively.

Lemma 3.30. *For $\delta \leq \frac{1}{4}$, given a δ-ONI instance with $|\mathcal{N}^1_1| \leq n(\frac{1}{4} - \delta)/(\frac{1}{4} + \frac{\delta}{3})$, all agents $i \in \mathcal{N}^1_1$ receive a bag of value at least $(3/4+\delta) \cdot \text{MMS}_i$ at the end of Algorithm 5.*

Input: δ-ONI $\mathcal{I} = (\mathcal{N}, \mathcal{M}, \mathcal{V})$ and factor δ
Output: Allocation $A = (A_1, \dots, A_n)$

Let $B_i = \{i, 2n - i + 1\}_{i \in [n]}$
Let $\mathcal{B} = \cup_{i \in [n]} \{B_i\}$
Let $\alpha = 3/4 + \delta$
while $\exists i \in \mathcal{N}, B \in \mathcal{B}$ s.t. $v_i(B) \geq \alpha$ **do**
    $i \leftarrow$ an arbitrary agent s.t. $v_i(B) \geq \alpha$, priority with agents in $\mathcal{N}_1^1$
    $A_i \leftarrow B$
    $\mathcal{B} \leftarrow \mathcal{B} \setminus \{B\}$
    $\mathcal{N} \leftarrow \mathcal{N} \setminus \{i\}$
    $\mathcal{M} \leftarrow \mathcal{M} \setminus B$
$J \leftarrow \cup_{B \in \mathcal{B}} B$
for $B \in \mathcal{B}$ **do**
    while $\nexists i \in \mathcal{N}$ s.t. $v_i(B) \geq \alpha$ **do**
        Let g be an arbitrary good in $\mathcal{M} \setminus J$
        $B \leftarrow B \cup \{g\}$
        $\mathcal{M} \leftarrow \mathcal{M} \setminus \{g\}$
    $i \leftarrow$ an arbitrary agent s.t. $v_i(B) \geq \alpha$, priority with agents in $\mathcal{N}_1^1$
    $A_i \leftarrow B$
    $\mathcal{N} \leftarrow \mathcal{N} \setminus \{i\}$
    $\mathcal{M} \leftarrow \mathcal{M} \setminus B$
return $(A_1, \dots, A_n)$

Algorithm 5: `approxMMS1`$(\mathcal{I}, \delta)$

Proof. It suffices to prove that all agents $i \in \mathcal{N}_1^1$ receive a bag at the end of Algorithm 5. Towards a contradiction, assume that $i \in \mathcal{N}_1^1$ does not receive any bag.

Claim 3.31. *For all bags B not allocated to an agent in $\mathcal{N}_1^1$, $v_i(B) < 3/4 + \delta$.*

Claim 3.31 holds since the priority is with agents in $\mathcal{N}_1^1$. Let S be the set of bags allocated to agents in $\mathcal{N}_1^1$ and $\bar{S}$ be the set of the remaining bags. We have

$$\begin{aligned} v_i(\mathcal{M}) &= \sum_{k \in [n]} v_i(\hat{B}_k) = \sum_{B \in S} v_i(B) + \sum_{B \in \bar{S}} v_i(B) \\ &< |\mathcal{N}_1^1| \left(1 + \frac{4\delta}{3}\right) + \left(n - |\mathcal{N}_1^1|\right) \left(\frac{3}{4} + \delta\right) && \text{(Lemma 3.11 and Claim 3.31)} \\ &\leq n, && (|\mathcal{N}_1^1| \leq n(\tfrac{1}{4} - \delta)/(\tfrac{1}{4} + \tfrac{\delta}{3})) \end{aligned}$$

which is a contradiction since $v_i(\mathcal{M}) = n$. Thus, all agents $i \in \mathcal{N}_1^1$ receive a bag at the end of Algorithm 5. $\square$

Remark 3.32. *The last inequality in the proof of Lemma 3.30 is tight for $|\mathcal{N}_1^1| = n(\frac{1}{4} - \delta)/(\frac{1}{4} + \frac{\delta}{3})$.*

Lemma 3.33. *For $\delta \leq \frac{1}{4}$, given a δ-ONI instance with $|\mathcal{N}_1^1| \leq n(\frac{1}{4} - \delta)/(\frac{1}{4} + \frac{\delta}{3})$, all agents $i \in \mathcal{N}_2^1$ receive a bag of value at least $(3/4 + \delta) \cdot \mathrm{MMS}_i$ at the end of Algorithm 5.*

Proof. It suffices to prove that all agents $i \in \mathcal{N}_2^1$ receive a bag at the end of Algorithm 5. Towards a contradiction, assume that $i \in \mathcal{N}_2^1$ does not receive any bag.

Claim 3.34. *For all $k \in [n]$, $v_i(\hat{B}_k) \leq 1$.*

Proof. The claim trivially holds if $\hat{B}_k = B_k$. Now assume $B_k \subsetneq \hat{B}_k$. Let g be the last good added to $\hat{B}_k$. We have $v_i(\hat{B}_k \setminus g) < 3/4 + \delta$, otherwise g would not be added to $\hat{B}_k$. Also note that $g \geq 2n+1$ and hence $v_i(g) \leq v_i(2n+1) < 1/4 - 5\delta$ by the definition of $\mathcal{N}_2^1$. Therefore, we have

$$\begin{aligned} v_i(\hat{B}_k) &= v_i(\hat{B}_k \setminus g) + v_i(g) \\ &< (\frac{3}{4} + \delta) + (\frac{1}{4} - 5\delta) < 1. \end{aligned}$$

Thus, the claim holds. ■

Since agent i did not receive a bag, there exists an unallocated bag with value less than 1 for agent i. Therefore, $v_i(\mathcal{M}) = \sum_{k\in[n]} v_i(\hat{B}_k) < n$ which is a contradiction. Thus, all agents $i \in \mathcal{N}_2^1$ receive a bag at the end of Algorithm 5. □

Agents in $\mathcal{N}^2$.

In this section, we prove that all agents in $\mathcal{N}^2$ also receive a bag at the end of Algorithm 5. For the sake of contradiction, assume that agent $i \in \mathcal{N}^2$ does not receive a bag at the end of Algorithm 5. Let $A^+ := \{k \in [n] \mid v_i(B_k) > 1\}$ and $A^- := \{k \in [n] \mid v_i(B_k) < 3/4 + \delta\}$ be the indices of the bags satisfying the respective constraint. Also, let ℓ be the smallest such that for all $k \in [\ell+1, n]$, $v_i(k) + v_i(2n - k + 1 + \ell) < 1$. See Figure 3.2.

Lemma 3.35. *For $\delta \leq 0.011$, given a δ-ONI instance with $|\mathcal{N}_1^1| \leq n(\frac{1}{4} - \delta)/(\frac{1}{4} + \frac{\delta}{3})$, all agents $i \in \mathcal{N}^2$ receive a bag of value at least $(\frac{3}{4} + \delta)$ at the end of Algorithm 5.*

Proof. It suffices to prove that all agents $i \in \mathcal{N}^2$ receive a bag at the end of Algorithm 5. Towards a contradiction, assume that $i \in \mathcal{N}^2$ does not receive any bag. For all $k \in \mathcal{N} \setminus (A^- \cup A^+)$, since $v_i(B_k) \geq 3/4 + \delta$ and i has not received a bag, $\hat{B}_k = B_k$. Thus, for all $k \in \mathcal{N} \setminus (A^- \cup A^+)$

$$v_i(\hat{B}_k) = v_i(B_k) \leq 1. \tag{3.2}$$

We have

$$\begin{aligned} n = v_i(\mathcal{M}) &= \sum_{k\in A^-} v_i(\hat{B}_k) + \sum_{k\in A^+} v_i(\hat{B}_k) + \sum_{k\in \mathcal{N}\setminus(A^-\cup A^+)} v_i(\hat{B}_k) \\ &< \left(|A^-|(\frac{5}{6} + 2\delta)\right) + \left(|A^+| + \ell(\frac{1}{12} + \delta)\right) + \left(n - |A^-| - |A^+|\right) \\ &\qquad \text{(Lemma 3.14, Lemma 3.15 and Inequality (3.2))} \\ &= n - |A^-|(\frac{1}{6} - 2\delta) + \ell(\frac{1}{12} + \delta). \end{aligned}$$

Therefore, we have

$$\frac{|A^-|}{\ell} < \frac{1/12 + \delta}{1/6 - 2\delta}. \tag{3.3}$$

Next, we bound the value of the goods in $\mathcal{M} \setminus [2n]$ and contradict Inequality (3.3). We have,

$$\begin{aligned}
\ell(\frac{1}{4} - \delta) &\leq v_i(\mathcal{M} \setminus [2n]) && \text{(Lemma 3.18)} \\
&= \sum_{k \in A^-} \Big(v_i(\hat{B}_k) - v_i(B_k) \Big) && (\mathcal{M} \setminus [2n] = \textstyle\bigcup_{k \in A^-} (\hat{B}_k \setminus B_k)) \\
&< |A^-| \left((\frac{5}{6} + \delta) - (\frac{1}{2} - 2\delta) \right) && \text{(Lemma 3.14 and Lemma 3.13)} \\
&= |A^-| \cdot (\frac{1}{3} + 3\delta).
\end{aligned}$$

Thus,

$$\frac{|A^-|}{\ell} > \frac{1/4 - \delta}{1/3 + 3\delta}. \tag{3.4}$$

Inequalities (3.3) and (3.4) imply that $\frac{1/12+\delta}{1/6-2\delta} > \frac{1/4-\delta}{1/3+3\delta}$, which is a contradiction with $\delta \leq 0.011$. Thus, all agents $i \in \mathcal{N}^2$ receive a bag at the end of Algorithm 5. ☐

Theorem 3.36. *Given any $\delta \leq 0.011$, for all δ-ONI instances where $|\mathcal{N}_1^1| \leq n(\frac{1}{4} - \delta)/(\frac{1}{4} + \frac{\delta}{3})$, Algorithm 5 returns a $(\frac{3}{4} + \delta)$-MMS allocation.*

Proof. Since $\mathcal{N} = \mathcal{N}_1^1 \cup \mathcal{N}_2^1 \cup \mathcal{N}^2$, by Lemmas 3.30, 3.33 and 3.35 all agents receive a bag of value at least $(\frac{3}{4} + \delta) \cdot \mathrm{MMS}_i$ in Algorithm 5. ☐

3.4.4 Case 2: $|\mathcal{N}_1^1| > n(\frac{1}{4} - \delta)/(\frac{1}{4} + \frac{\delta}{3})$

In this case, we run Algorithm 6. Starting from an ordered normalized $(3/4+\delta)$-irreducible instance, as long as there is a bag B_k with value at least $3/4 + \delta$ for some agent, we give B_k to such an agent. The priority is with agents who initially belonged to $\mathcal{N}_2^1 \cup \mathcal{N}^2$. Therefore, in the remaining instance, all bags are of value less than $3/4 + \delta$ for all the remaining agents. We introduce one more reduction rule in this section.

- R_5^α : If $v_i(1) + v_i(2) \geq \alpha$ for some $i \in \mathcal{N}$, allocate $\{1, 2\}$ to agent i and remove i from $\mathcal{N}$. The priority is with agents in $\mathcal{N}_2^1 \cup \mathcal{N}^2$.

Starting from an ordered normalized $(3/4 + \delta)$-irreducible instance, after allocating bags of value at least $3/4+\delta$ to some agents, we run $R_5^{3/4+\delta}$ as long as it is applicable. For ease of notation, we write R_j instead of $R_j^{3/4+\delta}$ for $j \in [5]$. Then, we run R_2 and R_3 as long as they are applicable. Afterwards, for all $k \in [n]$, we initialize $C_k = \{k, 2n-k+1, 2n+k\}$.[1] See Figure 3.3 for better intuition. Then, we do bag-filling. Let $\hat{C}_k$ be the result of bag-filling on bag C_k. The pseudocode of this algorithm is shown in Algorithm 6.

Lemma 3.37. *For all agents $i \in \mathcal{N}_2^1 \cup \mathcal{N}^2$ and bags B which is allocated to an agent in $\mathcal{N}_2^1 \cup \mathcal{N}^2$ during Algorithm 6, $v_i(B) < 3/2 + 2\delta$.*

[1] Note that it is without loss of generality to assume $m \geq 3n$. If $m < 3n$, add dummy goods of value 0 to everyone. The MMS value of the agents remains the same, and any α-MMS allocation in the final instance is an α-MMS allocation in the original instance after removing the dummy goods.

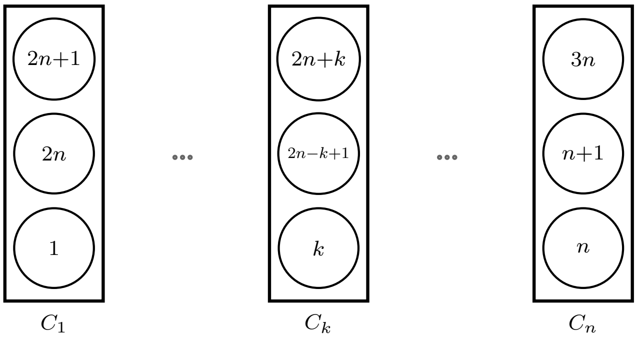


Figure 3.3: Configuration of Bags $C_1, C_2, \ldots, C_n$

Input: δ-ONI instance $\mathcal{I} = (\mathcal{N}, \mathcal{M}, \mathcal{V})$ and factor δ
Output: Allocation $A = (A_1, \ldots, A_n)$

Let $B_i = \{i, 2n - i + 1\}_{i \in [n]}$
Let $\mathcal{B} = \cup_{i \in [n]} \{B_i\}$
Let $\alpha = 3/4 + \delta$
while $\exists i \in \mathcal{N}, B \in \mathcal{B}$ s.t. $v_i(B) \geq \alpha$ **do**
  Let i be an arbitrary agent s.t. $v_i(B) \geq \alpha$, priority with agents in $\mathcal{N}_2^1 \cup \mathcal{N}^2$
  $A_i \leftarrow B$
  $\mathcal{B} \leftarrow \mathcal{B} \setminus \{B\}$
  $\mathcal{N} \leftarrow \mathcal{N} \setminus \{i\}$
  $\mathcal{M} \leftarrow \mathcal{M} \setminus B$
while $R_5^\alpha(\alpha)$ is applicable **do**
  apply $R_5^\alpha(\alpha)$
while R_2^α or R_3^α is applicable **do**
  apply R_k^α for smallest $k \in \{2, 3\}$ s.t. R_k^α is applicable
$n \leftarrow |\mathcal{N}|$
$C_i \leftarrow \{i, 2n - i + 1, 2n + i\}_{i \in [n]}$
for $k \leftarrow 1$ to n **do**
  while $\nexists i \in \mathcal{N}$ s.t. $v_i(C_k) \geq \alpha$ **do**
    Let g be an arbitrary good in $\mathcal{M} \setminus [3n]$
    $C_k \leftarrow C_k \cup \{g\}$
    $\mathcal{M} \leftarrow \mathcal{M} \setminus \{g\}$
  $i \leftarrow$ an arbitrary agent s.t $v_i(C_k) \geq \alpha$, priority with agents in $\mathcal{N}_2^1 \cup \mathcal{N}^2$
  $A_i \leftarrow C_k$
  $\mathcal{N} \leftarrow \mathcal{N} \setminus \{i\}$
  $\mathcal{M} \leftarrow \mathcal{M} \setminus C_k$
return $(A_1, \ldots, A_n)$

Algorithm 6: `approxMMS2`$(\mathcal{I}, \delta)$

Proof. We prove the lemma by upper bounding the value of the bags allocated at each step.

Claim 3.38. *For all bags B allocated to an agent before or during R_5, $v_i(B) < 3/2+2\delta$.*

Proof. Since we start with a $(3/4+\delta)$-irreducible instance, by Proposition 3.7, for all goods g, $v_i(g) < 3/4+\delta$. Therefore, for all the bags B of size two, we have $v_i(B) < 3/2+2\delta$. ■

Claim 3.39. *For all bags B which is allocated to an agent during R_2, $v_i(B) < 3/2+2\delta$.*

Proof. Note that when we run R_2, R_5 is not applicable. Therefore, $v_i(1)+v_i(2) < 3/4+\delta$. Hence, $v_i(\{2n-1, 2n, 2n+1\}) \leq v_i(\{1,2\}) + v_i(2n+1) < 2(3/4+\delta) = 3/2+2\delta$. ■

Claim 3.40. *For all bags B which is allocated to an agent during R_3, $v_i(B) < 3/2+2\delta$.*

Proof. Note that when we run R_3, R_5 is not applicable. Therefore, $v_i(1)+v_i(2) < 3/4+\delta$. Hence, $v_i(\{3n-2, 3n-1, 3n, 3n+1\}) \leq 2v_i(\{1,2\}) < 3/2+2\delta$. ■

Claim 3.41. *For all bags B allocated to an agent during the bag-filling phase, $v_i(B) < 3/2+2\delta$.*

Proof. If $B = \{k, 2n-k+1, 2n+k\}$, similar to the claims above, $v_i(B) \leq v_i(\{1,2\}) + v_i(2n+k) \leq 2(3/4+\delta) = 3/2+2\delta$. Otherwise, let g be the last good added to B. We have $v_i(B \setminus g) < 3/4+\delta$, otherwise g would not be added to B. Therefore, we have $v_i(B) = v_i(B \setminus g) + v_i(g) < 2(3/4+\delta) = 3/2+2\delta$. ■

By Claims 3.38, 3.39, 3.40 and 3.41, all bags that are allocated during Algorithm 6 are of value less than $3/2+2\delta$. Therefore, the lemma holds. □

Lemma 3.42. *For $\delta \leq 1/20$, given a δ-ONI instance with $|\mathcal{N}_1^1| > n(\frac{1}{4}-\delta)/(\frac{1}{4}+\frac{\delta}{3})$, all agents in $\mathcal{N}_2^1 \cup \mathcal{N}^2$ receive a bag of value at least $3/4+\delta$ at the end of Algorithm 6.*

Proof. It suffices to prove that all agents $i \in \mathcal{N}_2^1 \cup \mathcal{N}^2$ receive a bag at the end of Algorithm 6. Towards a contradiction, assume that $i \in \mathcal{N}_2^1 \cup \mathcal{N}^2$ does not receive any bag.

Claim 3.43. *For all bags B which is either unallocated or is allocated to an agent in $\mathcal{N}_1^1$, $v_i(B) < 3/4+\delta$.*

Proof. The claim holds since the priority is with agents in $\mathcal{N}_2^1 \cup \mathcal{N}^2$ and also that we allocate all the bags of value at least $3/4+\delta$ for some remaining agent. ■

Let S be the set of bags allocated to agents in $\mathcal{N}_2^1 \cup \mathcal{N}^2$ and $\bar{S}$ be the set of the remaining bags. We have

$$\begin{aligned}
n = v_i(\mathcal{M}) &= \sum_{B\in S} v_i(B) + \sum_{B\in\bar{S}} v_i(B) \\
&< (n - |\mathcal{N}_1^1|)\left(\frac{3}{2} + 2\delta\right) + |\mathcal{N}_1^1|\left(\frac{3}{4}+\delta\right) && \text{(Lemma 3.37 and Claim 3.43)} \\
&= \left(\frac{3}{4}+\delta\right)(2n - |\mathcal{N}_1^1|) \\
&< n(\frac{3}{4}+\delta)(2 - \frac{\frac{1}{4}-\delta}{\frac{1}{4}+\frac{\delta}{3}}). && (|\mathcal{N}_1^1| > n(\tfrac{1}{4}-\delta)/(\tfrac{1}{4}+\tfrac{\delta}{3})) \\
&= 3n(\frac{5\delta}{3}+\frac{1}{4})
\end{aligned}$$

This implies that $\frac{5\delta}{3} + \frac{1}{4} > \frac{1}{3}$. which is a contradiction with $\delta \leq 1/20$. Therefore, all agents $i \in \mathcal{N}_2^1 \cup \mathcal{N}^2$ receive a bag at the end of Algorithm 6. □

Agents in $\mathcal{N}_1^1$

Now we prove that all agents in $\mathcal{N}_1^1$ also receive a bag at the end of Algorithm 6. First, we prove a general lemma that lower bounds the MMS value of an agent after allocating $2k$ goods to k other agents. This way, we can lower bound the MMS value of agents in $\mathcal{N}_1^1$ after the sequence of R_5 rules is applied.

Lemma 3.44. *Given a set of goods $\mathcal{M}$ and a valuation function v, let $S \subseteq \mathcal{M}$ be such that $|S| = 2k$ for $k < n$ and let $x \geq 0$ be such that $v(g) \leq \mathrm{MMS}_v^n(\mathcal{M})/2 + x$ for all $g \in S$. Then, $\mathrm{MMS}_v^{n-k}(\mathcal{M} \setminus S) \geq \mathrm{MMS}_v^n(\mathcal{M}) - 2x$.*

Proof. We construct a partition of a subset of $\mathcal{M} \setminus S$ into $n-k$ bundles such that the minimum value of these bundles is at least $\mathrm{MMS}_v^n(\mathcal{M}) - 2x$. Let $(P_1, \ldots, P_n)$ be an MMS partition of $\mathcal{M}$ according to valuation function v. For all $j \in [n]$, let $Q_j = P_j \cap S$. Without loss of generality, assume $|Q_1| \geq \ldots \geq |Q_n|$. Let t be largest such that for all $\ell \leq t$, $\sum_{j\in[\ell]} |Q_j| \geq 2\ell$. This implies that $|Q_{t+1}| \leq 1$.

Claim 3.45. $\sum_{j\in[t]} |Q_j| = 2t$.

Proof. If $|Q_{t+1}| = 1$, and $\sum_{j\in[t]} |Q_j| > 2t$, then $\sum_{j\in[t+1]} |Q_j| \geq 2(t+1)$ which is a contradiction with the definition of t. If $|Q_{t+1}| = 0$, then $\sum_{j\in[t]} |Q_j| = 2k$. If $t < k$, then $\sum_{j\in[t+1]} |Q_j| = 2k \geq 2(t+1)$ which is again a contradiction with the definition of t. So in this case, $t = k$ and therefore, $\sum_{j\in[t]} |Q_j| = 2t$. Hence, Claim 3.45 holds. ■

Claim 3.46. $Q_{2k-t+1} = \emptyset$.

Proof. If $Q_{2k-t+1} \neq \emptyset$ then $|Q_{2k-t+1}| \geq 1$. Therefore,

$$\begin{aligned}
\sum_{j\in[2k-t+1]} |Q_j| &= \sum_{j\leq t} |Q_j| + \sum_{t<j\leq 2k-t+1} |Q_j| \\
&\geq 2t + (2k - 2t + 1) && \text{(Claim 3.45, and } |Q_j| \geq 1 \text{ for } j \leq 2k-t+1) \\
&> k,
\end{aligned}$$

which is a contradiction. Therefore, Claim 3.46 holds. ■

Now we remove the first t bundles (i.e., $P_1, \ldots, P_t$) and merge the next $k-t$ pairs of bundles after removing S (i.e., $(P_{t+1} \setminus S)$ with $(P_{t+2} \setminus S)$ and so on) as follows:

$$\hat{P} = \big((P_{t+1} \cup P_{t+2}) \setminus S, (P_{t+3} \cup P_{t+4}) \setminus S, \ldots, (P_{2k-t-1} \cup P_{2k-t}) \setminus S, P_{2k-t+1}, \ldots, P_n\big).$$

Claim 3.46 implies that for all $j > 2k-t$, $P_j = P_j \setminus S$. Therefore, $\hat{P}$ is a partition of the goods in $(\mathcal{M} \setminus (P_1 \cup \ldots \cup P_t)) \setminus S \subseteq \mathcal{M} \setminus S$. For all $j > 2k-t$, we have $v_i(P_j) \geq \text{MMS}_v^{|\mathcal{N}|}(\mathcal{M}) \geq \text{MMS}_v^{|\mathcal{N}|}(\mathcal{M}) - 2x$. Also, for all $t < j \leq 2k-t$, we have $|P_j \cap S| \leq 1$ and $v_i(g) \leq \text{MMS}_v^{|\mathcal{N}|}(\mathcal{M})/2 + x$ for all $g \in P_j$. Therefore,

$$\begin{aligned} v_i(P_j \setminus S) &\geq v_i(P_j) - (\text{MMS}_v^{|\mathcal{N}|}(\mathcal{M})/2 + x) \\ &\geq \text{MMS}_v^{|\mathcal{N}|}(\mathcal{M})/2 - x. \end{aligned}$$

Thus, for all $t < j < 2k-t$, $v_i((P_j \cup P_{j+1}) \setminus S) \geq \text{MMS}_v^{|\mathcal{N}|}(\mathcal{M}) - 2x$. Hence, $\hat{P}$ is a partition of a subset of $\mathcal{M} \setminus S$ into $n-k$ bundles with minimum value at least $\text{MMS}_v^{|\mathcal{N}|}(\mathcal{M}) - 2x$. Therefore, Lemma 3.44 holds. □

Lemma 3.47. *Let $i \in \mathcal{N}_1^1$ be a remaining agent after no more R_5 is applicable. Then, before applying more reduction rules,* $\text{MMS}_i \geq 1 - 12\delta$.

Proof. We start by proving the following claim.

Claim 3.48. *Right before applying any R_5, $v_i(1) \leq 1/2 + 6\delta$.*

Proof. Right before applying any R_5, no bag is of value at least $\frac{3}{4} + \delta$ to any agent and in particular agent i. Therefore, $v_i(1) + v_i(2n+1) \leq v_i(1) + v_i(2n) < 3/4 + \delta$. Since $v_i(2n+1) > 1/4 - 5\delta$, by the definition of R_5, we have $v_i(1) < 1/2 + 6\delta$. Therefore the claim holds. ■

Consider the step right before applying any R_5. Note that until this step, only some B_j's are allocated. Since $i \in \mathcal{N}_1^1$, $v_i(B_j) \leq 1$ for all $j \in [n]$ and since $|B_j| = 2$, allocating B_j's are valid reductions for agent i by Lemma 3.3. Thus, before applying any R_5, $\text{MMS}_i \geq 1$. Now let $\widehat{\mathcal{I}} = ([n'], \widehat{\mathcal{M}}, \mathcal{V})$ be the instance after applying the sequence of R_5's. Claim 3.48 and Lemma 3.44 imply that $\text{MMS}_{v_i}^{n'}(\widehat{\mathcal{M}}) \geq 1 - 12\delta$. □

For the sake of contradiction, assume that agent $i \in \mathcal{N}_1^1$ does not receive a bag at the end of Algorithm 6. By Lemma 3.47, $\text{MMS}_i \geq 1 - 12\delta$ after applying the sequence of R_5's. By Lemma 3.6, R_2 and R_3 are valid reductions for i and, therefore, $\text{MMS}_i \geq 1 - 12\delta$ at the beginning of the bag-filling phase. Let us abuse the notation and assume the instance at this step is $([n], [m], \mathcal{V})$.

Lemma 3.49. *Assuming $\delta \leq 1/212$, for all $k \in [n]$, if $v_i(C_k) \leq 1 - 12\delta$ for some agent i, then $v_i(\hat{C}_k) \leq 1 - 12\delta$.*

Proof. If $\hat{C}_k = C_k$, the claim follows. Otherwise, let g be the last good allocated to $\hat{C}_k$. We have $v_i(\hat{C}_k \setminus g) < 3/4 + \delta$, otherwise g would not be added to $\hat{C}_k$. Since $g > 3n$, by Proposition 3.7, $v_i(g) < 3/16 + \delta/4$. We have

$$\begin{aligned} v_i(\hat{C}_k) &= v_i(\hat{C}_k \setminus g) + v_i(g) \\ &< \left(\frac{3}{4} + \delta\right) + \left(\frac{3}{16} + \frac{\delta}{4}\right) \\ &= \frac{15}{16} + \frac{5\delta}{4} \leq 1 - 12\delta. && (\delta \leq 1/212) \end{aligned}$$

□

Lemma 3.50. *If $\delta \leq 1/212$, there exists $k \in [n]$ such that $v_i(C_k) > 1 - 12\delta$.*

Proof. For the sake of contradiction, assume that for all $k \in [n]$, $v_i(C_k) \leq 1 - 12\delta$. Since i did not receive a bag at the end of Algorithm 6, there exists an unallocated bag $\hat{C}_t$ such that $v_i(\hat{C}_t) < 3/4 + \delta$. We have

$$\begin{aligned} v_i(\mathcal{M}) = \sum_{k \in [n]} v_i(\hat{C}_k) &= \sum_{k \neq t} v_i(\hat{C}_k) + v_i(\hat{C}_t) \\ &< (n-1)(1 - 12\delta) + (\frac{3}{4} + \delta) && (\text{Lemma 3.49 and } v_i(\hat{C}_t) < \tfrac{3}{4} + \delta) \\ &< n(1 - 12\delta), && (\delta \leq 1/212) \end{aligned}$$

Note that $\text{MMS}_i \geq 1 - 12\delta$ and thus $v_i(\mathcal{M}) \geq n(1 - 12\delta)$ which is a contradiction and therefore, Lemma 3.50 holds. □

Let t be largest s.t. $v_i(C_t) > 1 - 12\delta$.

Observation 3.51. *Assuming $\delta \leq 1/212$, $t > 1$.*

Proof. For the sake of contradiction, assume $t = 1$. Since $1 - 12\delta \geq 3/4 + \delta$, we have

$$\begin{aligned} v_i(\hat{C}_1) = v_i(C_1) &= v_i(1) + v_i(2n) + v_i(2n+1) \\ &\leq v_i(1) + v_i(2) + (\frac{1}{4} + \frac{\delta}{3}) && (\text{Proposition 3.7}) \\ &< \left(\frac{3}{4} + \delta\right) + \left(\frac{1}{4} + \frac{\delta}{3}\right) = 1 + \frac{4\delta}{3}. && (R_5 \text{ is not applicable}) \end{aligned}$$

Also, since no bag is allocated to agent i, there must be a bag like C_ℓ with $v_i(\hat{C}_\ell) < \frac{3}{4} + \delta$.

$$\begin{aligned} n(1 - 12\delta) \leq v_i(\mathcal{M}) &= v_i(\hat{C}_1) + \sum_{k \in ([n] \setminus \{1, \ell\})} v_i(\hat{C}_k) + v_i(\hat{C}_\ell) \\ &< (1 + \frac{4\delta}{3}) + (n-2)(1 - 12\delta) + \frac{3}{4} + \delta && (\text{Lemma 3.49}) \\ &< n(1 - 12\delta), && (\delta \leq 1/212) \end{aligned}$$

which is a contradiction. Thus, $t > 1$. □

Observation 3.52. $v_i(2n+t) > 1/4 - 13\delta$.

Proof. We have

$$\begin{aligned}1 - 12\delta < v_i(C_t) &= v_i(t) + v_i(2n-t+1) + v_i(2n+t) \\ &\leq v_i(1) + v_i(2) + v_i(2n+t) && (t \geq 1 \text{ and } 2n-t+1 \geq 2) \\ &< \frac{3}{4} + \delta + v_i(2n+t). && (R_5 \text{ is not applicable})\end{aligned}$$

Therefore, $v_i(2n+t) > 1/4 - 13\delta$. □

Observation 3.53. $v_i(2n-t+1) > 3/8 - \delta(12 + 5/6)$.

Proof. Since R_5 is not applicable, $v_i(1)+v_i(2) < 3/4+\delta$ and therefore, $v_i(2) < 3/8+\delta/2$. We have

$$\begin{aligned}1 - 12\delta < v_i(C_t) &= v_i(t) + v_i(2n-t+1) + v_i(2n+t) && (C_t = \{t, 2n-t+1, 2n+t\}) \\ &\leq v_i(2) + v_i(2n-t+1) + (\frac{1}{4} + \frac{\delta}{3}) \\ & && (t \geq 2 \text{ by Observation 3.51 and } v_i(2n+t) < \tfrac{1}{4} + \tfrac{\delta}{3} \text{ by Proposition 3.7}) \\ &< (\frac{3}{8} + \frac{\delta}{2}) + v_i(2n-t+1) + (\frac{1}{4} + \frac{\delta}{3}) && (v_i(2) < \tfrac{3}{8} + \tfrac{\delta}{2}) \\ &= v_i(2n-t+1) + \frac{5}{8} + \frac{5\delta}{6}.\end{aligned}$$

Therefore, $v_i(2n-t+1) > 3/8 - \delta(12+5/6)$. □

Now let ℓ be largest such that $v_i(2n+\ell) \geq \delta(26+2/3)$.

Observation 3.54. *If $\delta \leq 3/476$, then $\ell \geq t$.*

Proof. By Observation 3.52, $v_i(2n+t) > 1/4 - 13\delta$. For $\delta \leq 3/476$, we have $1/4 - 13\delta \geq \delta(26+2/3)$. Thus, $\ell \geq t$. □

Lemma 3.55. *If $\delta \leq 3/956$, for all $k \leq \min(\ell, n)$, $v_i(C_k) \geq 3/4 + \delta$.*

Proof. By Observation 3.54, we have $\ell \geq t$. For all $k \leq t$ we have

$$\begin{aligned}v_i(C_k) &= v_i(k) + v_i(2n-k+1) + v_i(2n+k) && (C_k = \{k, 2n-k+1, 2n+k\}) \\ &\geq v_i(2n-t+1) + 2v_i(2n+t) \\ & && (k \leq 2n-t+1 \text{ and } 2n-k+1 < 2n+k \leq 2n+t) \\ &> \left(\frac{3}{8} - \delta(12 + \frac{5}{6})\right) + 2\left(\frac{1}{4} - 13\delta\right) && (\text{Observation 3.52 and 3.53}) \\ &= \frac{7}{8} - \delta(38 + \frac{5}{6}) \\ &\geq \frac{3}{4} + \delta. && (\delta \leq 3/956)\end{aligned}$$

Therefore, no good would be added to C_k for $k \leq t$. Now assume $t < k \leq \ell$. We have

$$\begin{aligned} v_i(C_k) &= v_i(k) + v_i(2n-k+1) + v_i(2n+k) && (C_k = \{k, 2n-k+1, 2n+k\}) \\ &\geq 2v_i(2n-t+1) + v_i(2n+\ell) \\ & && (k < 2n-k+1 < 2n-t+1 \text{ and } 2n+k \leq 2n+\ell) \\ &> 2\left(\frac{3}{8} - \delta(12+\frac{5}{6})\right) + \delta(26+\frac{2}{3}) && (\text{Observation 3.53 and the definition of } \ell) \\ &= \frac{3}{4} + \delta. \end{aligned}$$

□

Note that since i does not receive a bag by the end of Algorithm 6, there must be a remaining bag C_k such that $v_i(C_k) < 3/4 + \delta$. Thus, Lemma 3.55 implies that $\ell < n$ when $\delta \leq 3/956$.

Corollary 3.56 (of Lemma 3.55)**.** *If $\delta \leq 3/956$, for all $k \leq \ell$, $\hat{C}_k = C_k$.*

Observation 3.57. $v_i(\mathcal{M} \setminus \{1, 2, \ldots, 2n+\ell\}) \geq (n-\ell)(1/4 - 13\delta)$.

Proof. Consider the set of goods $\{1, 2, \ldots, 2n+\ell\}$ in the MMS partition of agent i. There exists $n-\ell$ bags in the MMS partition which have at most $2(n-\ell)$ goods in $\{1, 2, \ldots, 2n+\ell\}$ in total. Let P be the set of these bags. Let $G = \bigcup_{B \in P} B \cap \{1, 2, \ldots, 2n+\ell\}$. Since R_5 is not applicable, $v_i(\{a, b\}) < 3/4 + \delta$ for all distinct items a and b. Hence,

$$\begin{aligned} v_i(G) &\leq \lceil |G|/2 \rceil (3/4 + \delta) \\ &\leq (n-\ell)(3/4+\delta). && (|G| \leq 2(n-\ell)) \end{aligned}$$

Therefore,

$$\begin{aligned} v_i(\mathcal{M} \setminus \{1, 2, \ldots, 2n+\ell\}) &\geq v_i(\cup_{B \in P} B \setminus \{1, 2, \ldots, 2n+\ell\}) \\ &\geq (n-\ell)(1-12\delta) - (n-\ell)(3/4+\delta) \\ &= (n-\ell)(1/4 - 13\delta). \end{aligned}$$

□

Lemma 3.58. *If $\delta \leq 3/796$, for all $k > \ell$, $v_i(\hat{C}_k \setminus \{k, 2n-k+1\}) < 1/4 - 13\delta$.*

Proof. Since $1/4 - 13\delta \geq \delta(53 + 1/3)$ for $\delta \leq 3/796$, it suffices to prove $v_i(\hat{C}_k \setminus \{k, 2n-k+1\}) < \delta(53+1/3)$. Note that for all $k > \ell$, $v_i(2n+k) < \delta(26+2/3)$ by definition. Therefore, if $\hat{C}_k = C_k = \{k, 2n-k+1, 2n+k\}$, the lemma holds. Moreover, we have

$$\begin{aligned} v_i(\{k, 2n-k+1\}) &\geq 2v_i(2n-t+1) && (k < 2n-k+1 \leq 2n-t+1) \\ &> 2\left(\frac{3}{8} - \delta(12+\frac{5}{6})\right) && (\text{Observation 3.53}) \\ &= \frac{3}{4} - \delta(25+\frac{2}{3}). && (3.5) \end{aligned}$$

If $\hat{C}_k \neq C_k$, let g be the last good added to $\hat{C}_k$. Since $g > 3n + 1 > 2n + \ell$, $v_i(g) < \delta(26 + 2/3)$. We have $v_i(\hat{C}_k \setminus g) < 3/4 + \delta$ otherwise g would not be added to $\hat{C}_k$. We have

$$\begin{aligned} v_i(\hat{C}_k) &= v_i(\hat{C}_k \setminus g) + v_i(g) \\ &< \left(\frac{3}{4} + \delta\right) + \delta(26 + \frac{2}{3}) \\ &= \frac{3}{4} + \delta(27 + \frac{2}{3}). \end{aligned}$$

Hence,

$$\begin{aligned} \frac{3}{4} + \delta(27 + \frac{2}{3}) &> v_i(\hat{C}_k) \\ &= v_i(\{k, 2n - k + 1\}) + v_i(\hat{C}_k \setminus \{k, 2n - k + 1\}) \\ &> \frac{3}{4} - \delta(25 + \frac{2}{3}) + v_i(\hat{C}_k \setminus \{k, 2n - k + 1\}). \qquad \text{(Inequality (3.5))} \end{aligned}$$

Thus,

$$v_i(\hat{C}_k \setminus \{k, 2n - k + 1\}) < \delta(53 + \frac{1}{3}).$$

□

We are ready to prove Lemma 3.59.

Lemma 3.59. *For $\delta \leq 3/956$, given a δ-ONI instance with $|\mathcal{N}_1^1| > n(\frac{1}{4} - \delta)/(\frac{1}{4} + \frac{\delta}{3})$, all agents in $\mathcal{N}_1^1$ receive a bag of value at least $3/4 + \delta$ at the end of Algorithm 6.*

Proof. It suffices to prove that all agents $i \in \mathcal{N}_1^1$ receive a bag at the end of Algorithm 6. Towards a contradiction, assume that $i \in \mathcal{N}_1^1$ does not receive any bag. By Lemma 3.50, there exists a $k \in [n]$ such that $v_i(C_k) > 1 - 12\delta$. Recall that ℓ is largest such that $v_i(2n + \ell) \geq \delta(26 + 2/3)$. We have

$$\begin{aligned} (n - \ell)(\frac{1}{4} - 13\delta) &\leq v_i(\mathcal{M} \setminus \{1, 2, \ldots, 2n + \ell\}) && \text{(Observation 3.57)} \\ &= \sum_{k > \ell} v_i(\hat{C}_k \setminus \{k, 2n - k + 1\}) \\ & && (\hat{C}_k = C_k \text{ for } k \in [\ell] \text{ by Corollary 3.56}) \\ &< (n - \ell)(\frac{1}{4} - 13\delta), && \text{(Lemma 3.58)} \end{aligned}$$

which is a contradiction. □

Theorem 3.60. *Given any $\delta \leq 3/956$, for all δ-ONI instances where $|\mathcal{N}_1^1| > n(\frac{1}{4} - \delta)/(\frac{1}{4} + \frac{\delta}{3})$, Algorithm 6 returns a $(\frac{3}{4} + \delta)$-MMS allocation.*

Proof. For all other agents i, if $i \in \mathcal{N}_2^1 \cup \mathcal{N}^2$, by Lemma 3.42, i receives a bag of value at least $\frac{3}{4} + \delta$ and if $i \in \mathcal{N}_1^1$, by Lemma 3.59 i receives such a bag. Since $\mathcal{N} = \mathcal{N}_1^1 \cup \mathcal{N}_2^1 \cup \mathcal{N}^2$, the theorem follows. □

Input: Instance $\mathcal{I} = (\mathcal{N}, \mathcal{M}, \mathcal{V})$ and approximation factor $\alpha > 3/4$
Output: Allocation $A = (A_1, \dots, A_n)$

Let $\delta = 3/956$
$\mathcal{I} \leftarrow \texttt{order}(\texttt{normalize}(\texttt{reduce}_{\alpha}(\mathcal{I}))$
Let $\mathcal{N}_1^1 = \{i \in [n] \mid \forall j \in [n] : v_i(B_j) \leq 1 \text{ and } v_i(2n+1) \geq 1/4 - 5\delta\}$
if $|\mathcal{N}_1^1| \leq n(\frac{1}{4} - \delta)/(\frac{1}{4} + \frac{\delta}{3})$ **then**
  return `aprroxMMS1`$(\mathcal{I}, \delta)$ ▷ Algorithm 5 in Section 3.4.3
else
  return `aprroxMMS2`$(\mathcal{I}, \delta)$ ▷ Algorithm 6 in Section 3.4.4
return $(A_1, \dots, A_n)$

Algorithm 7: `mainApproxMMS`$(\mathcal{I}, \alpha)$

3.4.5 $(3/4 + \varepsilon)$-MMS allocations

In this section, we give the complete algorithm `mainApproxMMS`$(\mathcal{I}, \alpha)$ that achieves an α-MMS allocation for any instance $\mathcal{I}$ with additive valuations and $\alpha = 3/4 + \varepsilon$ for any $\varepsilon \leq 3/3836$. To this end, first we obtain a δ-ONI instance for $\delta = 4\varepsilon/(1-4\varepsilon)$ by running `order`(`normalize`(`reduce`$_{3/4+\varepsilon}(\mathcal{I})$)). Then depending on whether $|\mathcal{N}_1^1| \leq n(\frac{1}{4}-\delta)/(\frac{1}{4}+\frac{\delta}{3})$ or $|\mathcal{N}_1^1| > n(\frac{1}{4} - \delta)/(\frac{1}{4} + \frac{\delta}{3})$, we run `approxMMS1` or `approxMMS2`. The pseudocode of our algorithm `mainApproxMMS`$(\mathcal{I}, \alpha)$ is shown in Algorithm 7.

Theorem 3.61. *Given any instance $\mathcal{I} = (\mathcal{N}, \mathcal{M}, \mathcal{V})$ where agents have additive valuations and any $\alpha \leq \frac{3}{4} + \frac{3}{3836}$,* `mainApproxMMS`$(\mathcal{I}, \alpha)$ *returns an α-MMS allocation for $\mathcal{I}$.*

Proof. Let $\varepsilon = \alpha - 3/4$ and $\widehat{\mathcal{I}} =$ `order`(`normalize`(`reduce`$_{3/4+\varepsilon}(\mathcal{I})$)). Then by Theorem 3.24, $\widehat{\mathcal{I}}$ is ordered, normalized and $(\frac{3}{4} + \frac{4\varepsilon}{1-4\varepsilon})$-irreducible ($\frac{4\varepsilon}{1-4\varepsilon}$-ONI). Since $\varepsilon \leq \frac{3}{3836}$, $\frac{4\varepsilon}{1-4\varepsilon} \leq \frac{3}{956} = \delta$. Thus, $\widehat{\mathcal{I}}$ is δ-ONI. Furthermore, from any β-MMS allocation of $\widehat{\mathcal{I}}$ one can obtain a $\min(\frac{3}{4} + \varepsilon, (1-4\varepsilon)\beta)$-MMS allocation of $\mathcal{I}$.

By Theorem 3.36, given any $\delta \leq 3/956$, for all δ-ONI instances where $|\mathcal{N}_1^1| \leq n(\frac{1}{4} - \delta)/(\frac{1}{4} + \frac{\delta}{3})$, `approxMMS1` returns a $(\frac{3}{4} + \delta)$-MMS allocation. Also, by Theorem 3.60, for all δ-ONI instances where $|\mathcal{N}_1^1| > n(\frac{1}{4} - \delta)/(\frac{1}{4} + \frac{\delta}{3})$, `approxMMS2` returns a $(\frac{3}{4} + \delta)$-MMS allocation. Therefore, `mainApproxMMS`$(\mathcal{I}, \alpha)$ returns a $\min(\frac{3}{4} + \varepsilon, (1-4\varepsilon)(\frac{3}{4} + \delta))$-MMS allocation of $\mathcal{I}$. We have

$$\begin{aligned}(1-4\varepsilon)(\frac{3}{4} + \delta) &\geq (1 - \frac{3}{959})(\frac{3}{4} + \frac{3}{956}) \\ &= \frac{3}{4} + \frac{3}{3836} \\ &\geq \frac{3}{4} + \varepsilon = \alpha.\end{aligned}$$

Thus, `mainApproxMMS`$(\mathcal{I}, \alpha)$ returns an α-MMS allocation of $\mathcal{I}$. □

CHAPTER 4

Approximate MMS for XOS Valuations

In this chapter, we provide improved approximation guarantees for the maximin-share in the fractionally subadditive setting. We investigate randomized and deterministic allocation algorithms. Recall that in the XOS setting, for all agents i, there exists a family of additive valuation functions $u_{i,1}, u_{i,2}, \dots, u_{i,\ell} : 2^{\mathcal{M}} \to \mathbb{R}_{\geq 0}$, such that $v_i(S) = \max_{1 \leq k \leq \ell} u_{i,k}(S)$ for all subset of goods S.

Randomized Allocations

A randomized allocation is a distribution over deterministic allocations. A randomized allocation $\mathcal{R}$ has a property $\mathcal{P}$ ex-ante, if $\mathcal{P}$ holds on expectation. On the other hand, $\mathcal{R}$ has a property $\mathcal{P}$ ex-post, if all the allocations in the support of $\mathcal{R}$ have property $\mathcal{P}$. While having ex-ante guarantees might be easy in so many cases (i.e., being fair on average), the best-of-both-worlds idea is to have both ex-post and ex-ante guarantees simultaneously so that no matter which allocation in the support of $\mathcal{R}$ is realized, some fairness criterion is satisfied. For randomized allocations, we take one step toward extending the best-of-both-worlds idea for valuations more general than additive.

For the additive setting, [12] proved the existence of randomized allocations that are proportional ex-ante and 1/2-MMS ex-post. However, as we show in Section 4.2, though guaranteeing proportionality ex-ante is easy, for valuations such as submodular, XOS, and subadditive, this notion is not always a proper choice as a fairness criterion. In fact, for some instances, proportionality can be as small as $O(1/n)$ of the MMS value, which is highly undesirable. Therefore, here we focus on guaranteeing MMS approximations both ex-ante and ex-post. More precisely, we are looking for randomized allocations that are α-MMS ex-ante and β-MMS ex-post, where $0 < \beta < \alpha$.

In contrast to the additive setting, guaranteeing MMS ex-ante for XOS valuations is not easy. Note that in the additive setting, a fractional allocation that allocates a fraction $1/n$ of each item to each agent is proportional and consequently MMS. However, for some XOS instances this allocation is $O(1/n)$-MMS (Observation 4.15). Indeed, we show that there are instances for which guaranteeing MMS ex-ante is not possible. More precisely, we show that there are instances such that no randomized allocation can guarantee a factor better than 3/4 of her maximin-share to each agent (Lemma 4.16).

On the positive side, we propose an algorithm that finds a randomized allocation which is 1/4-MMS ex-ante.

Theorem 4.1. *For any instance with* XOS *valuations, there exists a randomized allocation that is* 1/4-MMS *ex-ante.*

Furthermore, by leveraging additional ideas, we extend this result to encompass both ex-ante and ex-post guarantees. Our proof for the approximation guarantee of our allocation is inspired by the work of [44]. In fact, we show that the fractional allocation $\mathcal{F}$ which is obtained from the following program is 1/4-MMS:

$$\begin{aligned} \text{maximize} \quad & \sum_{1\le i\le n} u_i & \\ \text{subject to} \quad & \sum_{1\le i\le n} f_{ij} = 1 & \forall_j \\ & f_{ij} \ge 0 & \forall_{i,j} \\ & u_i = \min(\frac{1}{2}, \max_k \sum_j u_{i,k}(g_j) f_{ij}). & \forall_i \end{aligned} \tag{4.1}$$

Intuitively, Program 4.1 defines an alternative valuation function $\bar{v}_i(\cdot)$ for each agent i and then finds an allocation that maximizes social welfare with respect to these valuations. For every i, $\bar{v}_i(\cdot)$ is the same as $v_i(\cdot)$, except that for bundles X with $v_i(X) > 1/2$ we have $\bar{v}_i(X) = 1/2$. From an economical standpoint, one can see the answer of this program as an interesting trade-off between fairness and social welfare.

We can then convert the fractional allocation $\mathcal{F}$ into a randomized one (Theorem 4.13 and Lemma 4.9). However, there is no non-trivial guarantee on the fairness of the ex-post allocation. To resolve this issue, we add one additional step to our algorithm (namely allocating single large items), and also add another constraint to the optimization program. Before solving the optimization problem, we check if a single item can satisfy an agent. The main goal of this step is to make sure that the value of each remaining item for the remaining agents is small enough so that we can use Theorem 4.13 to convert the fractional allocation into a randomized one with an ex-post guarantee.

We also add another constraint to the optimization problem to obtain the following half-integral optimization program:

$$\begin{aligned} \text{maximize} \quad & \sum_{1\le i\le n} u_i & \\ \text{subject to} \quad & \sum_{1\le i\le n} f_{ij} = 1 & \forall_j \\ & f_{ij} \in \{0, 1/2, 1\} & \forall_{i,j} \\ & u_i = \min(\frac{1}{2}, \max_k \sum_j u_{i,k}(g_j) f_{ij}). & \forall_i \end{aligned} \tag{4.2}$$

Despite this additional step and constraint, we prove that the answer of Program 4.2 also gives the 1/4-MMS ex-ante approximation guarantee. However, note that the upper bound on the value of items obtained from the additional step combined with Theorem 4.13 still gives no approximation guarantee better than 0 for the ex-post allocation, because there might be some items with value close to 1/4 in the bundle of agents. To prove the ex-post guarantee, we provide a more intricate analysis of the method that is used in Theorem 4.1 and show that our allocation is 1/8-MMS ex-post. The fact that the allocation is half-integral plays a key role in the proof. The pseudocode of our approach is shown in Algorithm 8.

Theorem 4.2. *For any instance with* XOS *valuations, Algorithm 8 returns a randomized allocation that is* 1/4-MMS *ex-ante and* 1/8-MMS *ex-post.*

Deterministic Allocations

Since the impossibility result on MMS by [62], there has been considerable work of establishing approximate MMS guarantees for various classes of valuation functions [13, 42, 44, 53, 63]. The best previous deterministic guarantee for XOS valuations was 0.2192235 [63]. We improve the guarantee to $3/13 \approx 0.2307$. In order to do so, like in our randomized algorithm, we first allocate large items. However, in addition to allocating single large items, here we also allocate pairs and triples of items if they satisfy some agent up to 3/13 factor of their MMS value. This way, we get a stronger upper bound on the value of most of the remaining items.

In the last step, we output an allocation which maximizes the social welfare with respect to valuations $\bar{v}_i(\cdot) = \min\{\frac{6}{13}, v_i(\cdot)\}$ for the remaining agents and items. Note that this last step is also in correspondence with the last step of the randomized algorithm. Here we cap the value of the bundles with 6/13 (instead of 1/2) and we output an integral allocation (instead of a half-integral allocation) with maximum social welfare. Moreover, here we need to do a more careful analysis to show that the output is indeed a 3/13-MMS allocation. The pseudocode of our approach is shown in Algorithm 9.

Theorem 4.3. *For any instance with* XOS *valuations, Algorithm 9 returns a* 3/13-MMS *allocation.*

4.1 Notations and Tools

Recall the definition of fractionally subadditive (XOS) valuations.

Definition 4.4 (XOS)**.** *A valuation function $v_i(\cdot)$ is fractionally subadditive (*XOS*), if there exists a family of additive valuation functions $u_{i,1}, u_{i,2}, \ldots, u_{i,\ell} : 2^{\mathcal{M}} \to \mathbb{R}_{\geq 0}$ such that for every set S we have*

$$v_i(S) = \max_{1 \leq k \leq \ell} u_{i,k}(S).$$

Given an allocation $\mathcal{A}$, we denote by $u_{i,i'}$, an additive function of v_i that defines $v_i(\mathcal{A}_i)$, i.e., $v_i(\mathcal{A}_i) = u_{i,i'}(\mathcal{A}_i)$. Another term we frequently use in this chapter is contribution; which is defined as the marginal value of one set to another.

Definition 4.5 (Contribution)**.** *For every sets S, T of items such that $S \subseteq T$, we define the marginal contribution of S to T with respect to valuation function v, denoted by $C_v^T(S)$, as follows:*

$$C_v^T(S) = v(T) - v(T \setminus S)$$

i.e., the marginal contribution of S to T is the value decrease when S is removed from T.

Example 4.6. *Consider 5 items $g_1, g_2, \ldots, g_5$, and an identical valuation v for all the agents. Suppose that v is a fractionally subadditive function consisting of two additive function $u_1 = [2, 8, 4, 5, 1]$ and $u_2 = [5, 1, 9, 4, 5]$ (the j^{th} element is the value* *for* g_j*). For set* $S = \{g_1, g_2, \ldots, g_5\}$ *we have* $u_1(S) = 20$ *and* $u_2(S) = 24$*. Hence,* $v(S) = \max(u_1(S), u_2(S)) = 24$*. Also, the marginal contribution of item* g_3 *to set* S *is* $C_v^S(\{g_3\}) = v(S) - v(S \setminus \{g_3\}) = 24 - 16 = 8$*, which is smaller than* $u_2(g_3) = 9$*.*

With abuse of notation, for an allocation $\mathcal{A}$ of items to agents with valuation vector $\mathcal{V} = (v_1, \ldots, v_n)$ and every set S of items, we define the contribution of S to $\mathcal{A}$ with respect to $\mathcal{V}$, denoted by $C_{\mathcal{V}}^{\mathcal{A}}(S)$ as follows:

$$C_{\mathcal{V}}^{\mathcal{A}}(S) = \sum_{1 \le i \le n} C_{v_i}^{\mathcal{A}_i}(\mathcal{A}_i \cap S). \tag{4.3}$$

However, since an XOS valuation function might include many additive functions, definition (4.3) is not always practical. Therefore, we use Observation 4.7 to bound $C_{\mathcal{V}}^{\mathcal{A}}(S)$. Since $u_{i,i'}(\mathcal{A}_i) = u_{i,i'}(\mathcal{A}_i \setminus S) + u_{i,i'}(\mathcal{A}_i \cap S)$ and we have $v_i(\mathcal{A}_i) - v_i(\mathcal{A}_i \setminus S) \le u_{i,i'}(\mathcal{A}_i) - u_{i,i'}(\mathcal{A}_i \setminus S) = u_{i,i'}(\mathcal{A}_i \cap S)$. Summing over i yields the following observation.

Observation 4.7. *For every allocation* $\mathcal{A}$ *of items to agents with valuation vector* $\mathcal{V}$ *and every set* S *of items, we have*

$$C_{\mathcal{V}}^{\mathcal{A}}(S) \le \sum_{1 \le i \le n} u_{i,i'}(\mathcal{A}_i \cap S),$$

for every i' *such that* $v(\mathcal{A}_i) = u_{i,i'}(\mathcal{A}_i)$*.*

For brevity, in the rest of the chapter, we assume that the valuations are scaled so that for every agent i, we have $\text{MMS}_i = 1$.

Randomized allocation. We also consider randomized allocations. A randomized allocation is a distribution over a set of deterministic allocations. For a randomized allocation $\mathcal{R}$, we denote by $D(\mathcal{R})$ the set of allocations in the support of $\mathcal{R}$. For a randomized allocation $\mathcal{R}$, the expected welfare of agent i is defined as

$$v_i(\mathcal{R}) = \sum_{\mathcal{A} \in D(\mathcal{R})} v_i(\mathcal{A}_i) \cdot p_{\mathcal{A}},$$

where $p_{\mathcal{A}}$ is the probability of allocation $\mathcal{A}$ in $\mathcal{R}$.

Fractional allocation. En route to proving our results, we leverage another relaxed form of allocation called fractional allocation. In a fractional allocation, we ignore the indivisibility assumption and treat each item as a divisible one. Formally, a fractional allocation $\mathcal{F}$ is a set of nm variables f_{ij} indicating the fraction of item g_j allocated to agent i. Therefore, a fractional allocation $\mathcal{F}$ satisfies the following constraints:

$$\forall_j \quad \sum_{1 \le i \le n} f_{ij} \le 1 \qquad \text{(we have one unit of each item)}$$

$$\forall_{i,j} \quad 0 \le f_{ij} \qquad \text{(each agent receives a non-negative share of each item)}$$

A fractional allocation is *complete* if $\sum_i f_{ij} = 1$ for all items g_j, i.e., all items are completely allocated. Given a fractional allocation $\mathcal{F}$, we define the utility of agent i for $\mathcal{F}$ in the same way as we calculate it for integral allocations:

$$v_i(\mathcal{F}) = \max_k \sum_{1 \le j \le m} u_{i,k}(g_j) f_{ij}.$$

Complete fractional allocations give rise to randomized allocations in the standard way, i.e., the probability $p_{\mathcal{A}}$ of an allocation $\mathcal{A}$ is defined as

$$p_{\mathcal{A}} = \prod_{1 \le i \le n} \prod_{g_j \in \mathcal{A}_i} f_{ij}. \tag{4.4}$$

Then $\sum_{\mathcal{A}} p_{\mathcal{A}} = 1$. Indeed, one can view an integral allocation $\mathcal{A}$ as a mapping π from $[m]$ to $[n]$: $\pi(j) = i$ iff $g_j \in \mathcal{A}_i$. Then $\sum_{\mathcal{A}} p_{\mathcal{A}} = \sum_{\mathcal{A}} \prod_i \prod_{g_j \in \mathcal{A}_i} f_{ij} = \sum_{\pi \in [n]^{[m]}} \prod_j f_{\pi(j)j} = \prod_j (\sum_i f_{ij}) = \prod_j 1 = 1$. Also, we prove in the next lemma the probability that a particular agent i receives a particular set S is $\prod_{j \in S} f_{ij} \prod_{j \notin S} (1 - f_{ij})$.

Lemma 4.8. *Let $\mathcal{F}$ be a complete fractional allocation and let randomized allocation $\mathcal{R}$ be defined by (4.4). Then for all sets $S \subseteq \mathcal{M}$ and agents i,*

$$P[\mathcal{A}_i = S] = \prod_{j \in S} f_{ij} \prod_{j \notin S} (1 - f_{ij}).$$

Proof.

$$\begin{aligned}
P[\mathcal{A}_i = S] &= \sum_{\mathcal{A}; \mathcal{A}_i = S} p_{\mathcal{A}} \\
&= \sum_{\mathcal{A}; \mathcal{A}_i = S} \prod_{1 \le \ell \le n} \prod_{j \in \mathcal{A}_\ell} f_{\ell j} \\
&= \prod_{j \in S} f_{ij} \sum_{\mathcal{A}; \mathcal{A}_i = S} \prod_{\ell \ne i} \prod_{j \in \mathcal{A}_\ell} f_{\ell j} \\
&= \prod_{j \in S} f_{ij} \prod_{j \notin S} \sum_{\ell \ne i} f_{\ell j} \\
&= \prod_{j \in S} f_{ij} \prod_{j \notin S} (1 - f_{ij}).
\end{aligned}$$

□

Next, we prove that all the agents value the randomized allocation $\mathcal{R}$ obtained by the fractional allocation $\mathcal{F}$, at least as much as $\mathcal{F}$. The intuition is as follows. For all agents i, in order to compute $v_i(\mathcal{F})$ we select the best function $u_{i,i'}$ for the fractional set which contains a fraction f_{ij} of item g_j for each j. In $v_i(\mathcal{R})$ on the other hand, we form a weighted sum over integral sets (the weight of a set S is $\prod_{j \in S} f_{ij} \prod_{j \notin S} (1 - f_{ij})$) and choose the best function for each set.

Lemma 4.9. *Let $\mathcal{F}$ be a complete fractional allocation and let randomized allocation $\mathcal{R}$ be defined by (4.4). Then for* XOS *valuation functions v_i,*

$$v_i(\mathcal{R}) \ge v_i(\mathcal{F})$$

for all $1 \le i \le n$.

Proof. For each i, let i' be such that $u_{i,i'}(\mathcal{F}) = \max_k u_{i,k}(\mathcal{F})$.

$$\begin{aligned}
v_i(\mathcal{R}) &= \sum_S \sum_{\mathcal{A};\mathcal{A}_i=S} v_i(S)p_{\mathcal{A}} \\
&= \sum_S v_i(S) \prod_{j\in S} f_{ij} \prod_{j\notin S}(1-f_{ij}) && \text{(Lemma 4.8)} \\
&\geq \sum_S u_{i,i'}(S) \prod_{j\in S} f_{ij} \prod_{j\notin S}(1-f_{ij}) \\
&= \sum_S \left(\sum_{j\in S} u_{i,i'}(j)\right) \prod_{j\in S} f_{ij} \prod_{j\notin S}(1-f_{ij}) \\
&= \sum_j u_{i,i'}(j) f_{ij} \sum_{S;j\in S} \prod_{\ell\in S;\ell\neq j} f_{ij} \prod_{\ell\notin S}(1-f_{i\ell}) \\
&= \sum_j u_{i,i'}(j) f_{ij} \prod_{\ell\neq j}(f_{i\ell}+1-f_{i\ell}) \\
&= \sum_j u_{i,i'}(j) f_{ij} \\
&= v_i(\mathcal{F}).
\end{aligned}$$

□

We also need to define *contribution* for fractional allocations and fractional bundles. Suppose that $\mathcal{F}$ is a fractional allocation and S is a fractional set of items. Since items are fractionally allocated, the term contribution must be defined more precisely. For example, suppose that set S consists of a fraction 0.4 of item g_j, and in allocation $\mathcal{F}$, 0.2 of g_j belongs to agent i_1, 0.5 of g_j belongs to agent i_2, and 0.3 of g_j belongs to agent i_3. We need to define exactly how the 0.4 fraction of item g_j in S is distributed over the agents. One reasonable strategy is to choose the share of each agent in a way that after removal of S from $\mathcal{F}$ we have the smallest possible decrease in the social welfare. Based on this strategy, assuming that s_j is the fraction of item g_j in S, we define the contribution of S to $\mathcal{F}$, denoted by $C^{\mathcal{F}}_{\mathcal{V}}(S)$ as the value of the following optimization program:

$$\begin{aligned}
\text{minimize} \quad & \sum_{1\leq i\leq n} (v_i(\mathcal{F}) - v_i(\mathcal{F}')) \\
\text{subject to} \quad & \sum_{1\leq i\leq n} f_{ij} - f'_{ij} = s_j && \forall_j \\
& 0 \leq f'_{ij} \leq f_{ij} && \forall_{i,j} \qquad (4.5)
\end{aligned}$$

Generally, it is hard to deal with the above optimization program. Here, we use an important property of $C^{\mathcal{F}}_{\mathcal{V}}(\cdot)$ to obtain our results.

Lemma 4.10. *Let $\mathcal{F}$ be an arbitrary fractional allocation and assume that for every agent i, $v_i(\cdot)$ is* XOS*. Then, for every partition of the items into fractional sets $S_1, S_2, \ldots, S_t$, we have*

$$\sum_{1\leq k\leq t} C^{\mathcal{F}}_{\mathcal{V}}(S_k) \leq \sum_{1\leq i\leq n} v_i(\mathcal{F});$$

i.e, the sum of the contributions cannot exceed the total value of $\mathcal{F}$.

Proof. For every agent i, let $i' = \arg\max_k u_{i,k}(\mathcal{F})$. By definition, we have

$$v_i(\mathcal{F}) = \sum_{1 \le j \le m} u_{i,i'}(g_j) \cdot f_{ij}.$$

We will define for each set S_k an allocation $\mathcal{F}^{(k)}$ by reducing the allocation $\mathcal{F}$ proportionally, i.e., we will replace f_{ij} by $(1 - s_{k,j})f_{ij}$, where $s_{k,j}$ is the fraction of item j belonging to set S_k. Note that since $S_1, S_2, \ldots, S_t$ is a partition of items, $\sum_k s_{k,j} = 1$. For every $1 \le k \le t$, $1 \le i \le n$, and $1 \le j \le m$, define variable $f'^{(k)}_{ij}$ as follows:

$$f'^{(k)}_{ij} = (1 - s_{k,j}) \cdot f_{ij}.$$

Denote by $\mathcal{F}'^{(k)}$ the partial allocation defined by the variables $f'^{(k)}_{ij}$. Since for every j, we have

$$\begin{aligned}\sum_{1 \le i \le n} (f_{ij} - f'^{(k)}_{ij}) &= \sum_{1 \le i \le n} s_{k,j} f_{ij} \\ &= s_{k,j},\end{aligned}$$

$\mathcal{F}'^{(k)}$ is a feasible solution to Program (4.5). Therefore, we have

$$\begin{aligned}\sum_{1 \le k \le t} C^{\mathcal{F}}_{\mathcal{V}}(S_k) &\le \sum_{1 \le k \le t} \sum_{1 \le i \le n} (v_i(\mathcal{F}) - v_i(\mathcal{F}'^{(k)})) \\ &\le \sum_{1 \le k \le t} \sum_{1 \le i \le n} \sum_{1 \le j \le m} u_{i,i'}(g_j)(f_{ij} - f'^{(k)}_{ij}) \\ &= \sum_{1 \le i \le n} \sum_{1 \le j \le m} \sum_{1 \le k \le t} u_{i,i'}(g_j) s_{k,j} f_{ij} \\ &= \sum_{1 \le i \le n} \sum_{1 \le j \le m} u_{i,i'}(g_j) f_{ij} \\ &= \sum_{1 \le i \le n} v_i(\mathcal{F}).\end{aligned}$$

□

4.1.1 Ex-ante and Ex-post Fairness Guarantees.

For a randomized allocation, we define two types of fairness guarantees, namely *ex-ante* and *ex-post* as in Definitions 4.11 and 4.12.

Definition 4.11 (ex-ante)**.** *Given a randomized allocation $\mathcal{R}$, we say $\mathcal{R}$ is α-*MMS ex-ante*, if for every agent i, we have $v_i(\mathcal{R}) \ge \alpha \cdot \mathrm{MMS}_i$. Similarly, $\mathcal{R}$ is α-proportional, if for every agent i, $v_i(\mathcal{R}) \ge \alpha \cdot \pi_i$.*

Definition 4.12 (ex-post)**.** *An allocation $\mathcal{R}$ is α-*MMS ex-post*, if every allocation $\mathcal{A} \in D(\mathcal{R})$ is α-*MMS*. Similarly, we say $\mathcal{R}$ is α-proportional ex-post if every allocation $\mathcal{A} \in D(\mathcal{R})$ is α-proportional.*

Babaioff, Ezra, and Feige [12] obtain how to convert a fractional allocation into a faithful randomized allocation.

Theorem 4.13 ([12]). *Assume that the valuations are additive and let $\mathcal{F}$ be a fractional allocation. Then there exists a randomized allocation $\mathcal{R}$ such that the ex-ante utility of the agents for $\mathcal{R}$ is the same as the utility of the agents in $\mathcal{F}$, and for every allocation $\mathcal{A}$ in the support of $\mathcal{R}$ the following holds:*

$$\forall i : v_i(\mathcal{A}_i) \geq v_i(\mathcal{R}) - \max_{j: f_{ij} \notin \{0,1\}} v_i(g_j).$$

We will use a more delicate analysis of the method used in Theorem 4.13 to convert fractional allocations into randomized ones. This helps us improve our ex-post approximation guarantee.

4.2 Ex-ante Guarantees

In this section, our goal is to design randomized allocations that is α-MMS ex-ante or α-proportional ex-ante. Note that, in contrast to the additive case, for XOS valuations there is no meaningful correspondence between proportionality and maximin-share; the proportional share can be larger or smaller than the maximin-share. Recall that for the additive case, we always have $\pi_i \geq \mathrm{MMS}_i$ (Observation 2.4) and therefore, maximin-share is implied by proportionality. However, for fractionally subadditive valuations, π_i can be as small as MMS_i/n.

For the additive setting, a simple fractional allocation that allocates a fraction $1/n$ of each item to each agent guarantees proportionality and consequently maximin-share. Using Theorem 4.13 one can convert this allocation to a randomized allocation that is proportional ex-ante. In Observation 4.14 we show that proportionality can be guaranteed ex-ante for XOS valuations.[1]

Observation 4.14. *Every randomized allocation that allocates each item with probability $1/n$ to each agent is proportional ex-ante.*

Proof. Let $\mathcal{R}$ be a randomized allocation such that the probability that item g_j is allocated to agent i is $1/n$. Also, let $u_{i,i'}$ be the additive valuation function that defines $v_i(\mathcal{M})$, i.e.,

[1]Note that for now, we are not concerned about the ex-post guarantee of our allocation.

$v_i(\mathcal{M}) = u_{i,i'}(\mathcal{M})$, and let $p_\mathcal{A}$ be the probability that allocation $\mathcal{A}$ is chosen in $\mathcal{R}$. We have

$$\begin{aligned}
v_i(\mathcal{R}) &= \sum_{\mathcal{A} \in D(\mathcal{R})} p_\mathcal{A} \cdot v_i(\mathcal{A}_i) \\
&\geq \sum_{\mathcal{A} \in D(\mathcal{R})} p_\mathcal{A} \cdot u_{i,i'}(\mathcal{A}_i) \\
&= \sum_{\mathcal{A} \in D(\mathcal{R})} p_\mathcal{A} \sum_{g_j \in \mathcal{A}_i} u_{i,i'}(g_j) \\
&= \sum_{1 \leq j \leq m} \sum_{i : g_j \in \mathcal{A}_i} u_{i,i'}(g_j) \cdot p_\mathcal{A} \\
&= \sum_{1 \leq j \leq m} u_{i,i'}(g_j)/n \\
&= v_i(\mathcal{M})/n.
\end{aligned}$$

□

In contrast to the additive setting, finding a randomized allocation that guarantees the maximin-share ex-ante is not trivial. Indeed, the simple fractional allocation that guarantees proportionality in Observation 4.14 can be as bad as $O(1/n)$-MMS.

Observation 4.15. *Let $\mathcal{F}$ be a fractional allocation that allocates a fraction $1/n$ of each item to each agent. Then, there exists an instance such that the approximate maximin-share guarantee of $\mathcal{F}$ is $1/n$.*

Proof. Consider the following instance. There are n^2 items. The valuation of agent i is an XOS set function consisting of n additive valuation functions as follows: partition the items into n bundles each with n items. For each additive function $u_{i,k}$, the value of each item in the k^{th} bundle is $1/n$ and the value of the rest of the items is 0. It is easy to observe that for this instance, the MMS value of each agent is 1, and the value of each agent for her bundle in $\mathcal{F}$ is $1/n$. □

Generally, there are two main challenges in the process of designing a randomized allocation that guarantees an approximation of the maximin-share. In contrast to the additive setting, finding a fractional or randomized allocation that approximates maximin-share is not easy. Also, transforming a fractional allocation into a randomized one is not straightforward. Indeed, as we show in Lemma 4.16, neither fractional allocations nor randomized allocations can guarantee MMS. We prove an upper bound on the best approximation guarantee of each one of these allocation types.

Lemma 4.16. *For* XOS *valuations, the best approximate* MMS *guarantee for fractional allocations and the best ex-ante approximate* MMS *guarantee for randomized allocation is upper bounded by* $3/4$.

Proof. Consider the following instance: there are two agents and four items. The fractionally subadditive valuation of each agent consists of 2 additive functions. The valuations are as follows:

$$\begin{aligned}
u_{1,1}(\{g_1\}) = u_{1,1}(\{g_2\}) = 1, u_{1,1}(\{g_3\}) = u_{1,1}(\{g_4\}) = 0,\\
u_{1,2}(\{g_1\}) = u_{1,2}(\{g_2\}) = 0, u_{1,2}(\{g_3\}) = u_{1,2}(\{g_4\}) = 1,\\
u_{2,1}(\{g_1\}) = u_{2,1}(\{g_4\}) = 1, u_{2,1}(\{g_2\}) = u_{2,1}(\{g_3\}) = 0,\\
u_{2,2}(\{g_1\}) = u_{2,2}(\{g_4\}) = 0, u_{2,2}(\{g_2\}) = u_{2,2}(\{g_3\}) = 1.
\end{aligned}$$

It is easy to check that for the above instance, the maximin-share of each agent is equal to 2, and no fractional allocation can guarantee more than 1.5 to both agents. Also, we can guarantee a value of 1.5 to both agents by giving the first and the third item respectively to agents 1 and 2, and giving half of the remaining items to each agent.

Now, we show that the same upper bound also holds for the ex-ante guarantee of randomized allocations. Assume that $\mathcal{R}$ is the randomized allocation that maximizes the maximin-share guarantee for this instance. Since there are two agents, we know that $\mathcal{R}$ maximizes the following objective:

$$\alpha = \min\left(\sum_{S\subseteq\mathcal{M}} \mathbb{P}(S)v_1(S), \sum_{S\subseteq\mathcal{M}} \mathbb{P}(S)v_2(\mathcal{M}\setminus S)\right),$$

where $\mathbb{P}(S)$ is the probability that set S is allocated to agent 1. Since for all integers y, z we have $\min(y, z) \le (y + z)/2$, we obtain

$$\begin{aligned}
\alpha &\le \left(\sum_{S\subseteq\mathcal{M}} \mathbb{P}(S)v_1(S) + \sum_{S\subseteq\mathcal{M}} \mathbb{P}(S)v_2(\mathcal{M}\setminus S)\right)/2\\
&= \sum_{S\subseteq\mathcal{M}} \mathbb{P}(S)\left(v_1(S) + v_2(\mathcal{M}\setminus S)\right)/2.
\end{aligned}$$

One can easily check that for every set S, the value of $v_1(S)+v_2(\mathcal{M}\setminus S)$ is upper bounded by 3. Therefore, we have

$$\begin{aligned}
\alpha &\le \sum_{S\subseteq\mathcal{M}} (3/2)\mathbb{P}(S)\\
&\le 3/2.
\end{aligned}$$

Hence, the best possible approximation guarantee for MMS in this instance is at most $1.5/2 = 3/4$. Note that one can guarantee a value of 1.5 ex-ante to both agents by giving the first and the third item respectively to agents 1 and 2, and giving the other two items items with a probability of 1/2 to each of the agents. □

Before we prove our lower bound on the maximin-share guarantee for randomized allocations, we note that another challenge about XOS valuations is that in sharp contrast to additive valuations, transforming a fractional allocation to a randomized one is not easy. Indeed, we can show that for a fractional allocation $\mathcal{F}$ there might be randomized allocations $\mathcal{R}$ and $\mathcal{R}'$ with different utility guarantees for the agents, such that in both $\mathcal{R}$ and $\mathcal{R}'$ the probability that each item g_j is allocated to agent i is equal to f_{ij}. Example 4.17 gives more insight into this challenge.

Example 4.17. *Consider the instance described in the proof of Observation 4.15 and define allocations $\mathcal{R}$ and $\mathcal{R}'$ as follows:*

- *Allocation $\mathcal{R}$ allocates each item to each agent with probability $1/n$.*
- *Allocation $\mathcal{R}'$ considers a random permutation of the bundles in the optimal* MMS*-partitioning of the agents and allocates the i^{th} bundle in the permutation to agent i.*[2]

It is easy to check that in both of these allocations, each item is allocated to each agent with probability $1/n$. However, the approximate maximin-share guarantee of $\mathcal{R}$ is $O(\log n/n)$. To show this, one can argue that using Chernoff bound the probability that more than $3\log n$ items from the same bundle in the optimal partition are allocated to agent i is $O(1/n^2)$. Hence, the expected value of agent i for her share is at most

$$1 \cdot \frac{1}{n^2} + \frac{3\log n}{n} \cdot \frac{n^2-1}{n^2} \leq \frac{4\log n}{n}.$$

On the other hand, allocation $\mathcal{R}'$ guarantees value 1 to all the agents.

Despite these hurdles, in Theorem 4.1 we show that there exists a randomized allocation that is 1/4-MMS ex-ante. To prove Theorem 4.1, we first show that a fractional allocation exists that is 1/4-MMS. Next, we convert it to a randomized allocation. Theorem 4.1 along with Lemma 4.16 leave a gap of $[1/4, 3/4)$ between the best upper bound and the best lower-bound for the approximate maximin-share guarantee of randomized allocations in the XOS setting.

Theorem 4.1. *For any instance with* XOS *valuations, there exists a randomized allocation that is* 1/4-MMS *ex-ante.*

Proof Idea: Let $\mathcal{F}$ be a (fractional) allocation maximizing social welfare and assume that there is an agent i^* whose bundle has value less than 1/4 to her. Now consider the MMS-partition of agent i^*. Each bundle in this partition has value at least 1 to i^*. We can split each bundle into two so that each sub-bundle has value at least 1/2 to i^*. Let B be one of these $2n$ sub-bundles. Imagine that we reassign the items in B. We take away the items in B from their current owners and give them to i^*. Then i^* would gain more than 1/4, but the other agents would lose. The loss is bounded by $C_{\mathcal{F}}(B)$. Why should this quantity be less than 1/4 for one of the $2n$ sub-bundles?

Lemma 4.10 comes to the rescue. We have

$$\sum_{1\leq j\leq 2n} C^{\mathcal{F}}_{\mathcal{V}}(B_j) \leq \sum_{1\leq i\leq n} v_i(\mathcal{F}),$$

where B_1 to B_{2n} are the sub-bundles. If the right hand side is strictly less than $n/2$, the desired sub-bundle exists. We can achieve this by replacing our valuations v_i by valuations $\bar{v}_i$ that assign no set a value more than 1/2.

[2]Note that in the instance described in Observation 4.15 the optimal MMS-partitioning of all the agents are the same.

Proof. For a fractional allocation, we define the truncated value of agent i, denoted by $\overline{v}_i$, of a fractional set S as follows:

$$\overline{v}_i(S) = \min\left(\frac{1}{2}, \max_k \sum_{1\leq j\leq m} u_{i,k}(g_j)s_j\right). \tag{4.6}$$

where $u_{i,k}$ is the k^{th} additive function of v_i and s_j is the fraction of item g_j that belongs to set S. If the valuation $v_i(\cdot)$ is XOS, then $\bar{v}_i(\cdot)$ is also XOS [44]. Let $\bar{\mathcal{V}} = (\bar{v}_1, \ldots, \bar{v}_n)$.

Now let $\mathcal{F}$ be the complete fractional allocation that maximizes

$$Z = \sum_{1\leq i\leq n} \bar{v}_i(\mathcal{F}). \tag{4.7}$$

We know $Z \leq n/2$ since for any fractional bundle S and every agent i, we have $\overline{v}_i(S) \leq 1/2$. We claim that $\mathcal{F}$ allocates each agent a bundle with a value of at least 1/4. For the sake of a contradiction, assume that this is not true and let agent i^* be an agent whose share is worth less than 1/4 to her. Then $Z < n/2 - 1/4$.

Claim 4.18. *For all agents i, since the maximin-share of i is at least 1, she can divide the items in $\mathcal{M}$ into $2n$ fractional bundles, each with value at least 1/2 to her.*

Proof. Consider the optimal maximin-share partition of agent i, and for each bundle in this partition divide that bundle into two fractional sub-bundles with a value of at least 1/2. Since the valuation of agent i is XOS, such a division is always possible: just take a fractional sub-bundle with a value of exactly 1/2 from each bundle. The remaining (fractional) items in that bundle also form a sub-bundle with a value of at least 1/2. ■

Let $B_1, B_2, \ldots, B_{2n}$ be these $2n$ bundles. By applying Lemma 4.10 we have

$$\sum_{1\leq j\leq 2n} C^{\mathcal{F}}_{\bar{\mathcal{V}}}(B_j) \leq \sum_{1\leq i\leq n} \bar{v}_i(\mathcal{F}) = Z.$$

Note that here $C^{\mathcal{F}}_{\bar{\mathcal{V}}}$ refers to the contribution with respect to $(\bar{v}_1, \ldots, \bar{v}_n)$. Therefore, at least one of the bundles contributes less than $Z/(2n) < 1/4$ to Z. Let B_k be one such bundle, i.e., $C^{\mathcal{F}}_{\bar{\mathcal{V}}}(B_k) < 1/4$. Let b_{kj} be the fraction of item g_j belonging to bundle B_k and let $\mathcal{F}'$ be the allocation that defines the contribution of bundle B_k to allocation $\mathcal{F}$ (see Program 4.5). Then $\sum_i f'_{ij} = \sum_i f_{ij} - b_{kj}$ for all j and

$$\sum_{1\leq i\leq n} \bar{v}_i(\mathcal{F}) - \sum_{1\leq i\leq n} \bar{v}_i(\mathcal{F}') = C^{\mathcal{F}}_{\bar{\mathcal{V}}}(B_k) < 1/4,$$

which means

$$\sum_{1\leq i\leq n} \bar{v}_i(\mathcal{F}') > Z - \frac{1}{4}.$$

We now assign the items in B_k to agent i^*, i.e., we consider the fractional allocation $\mathcal{F}''$ equal to $\mathcal{F}'$, except that for agent i^*, we have

$$f''_{i^*j} = f'_{i^*j} + b_{kj} \qquad \text{for all } j \in [1 \ldots m]$$

Since the value of bundle B_k to agent i^* is at least 1/2, we have $\bar{v}_i(\mathcal{F}'') - \bar{v}_i(\mathcal{F}') > 1/4$, and further

$$\sum_{1\le i\le n} \bar{v}_i(\mathcal{F}'') > \sum_{1\le i\le n} \bar{v}_i(\mathcal{F}') + \frac{1}{4} > (Z - \frac{1}{4}) + \frac{1}{4} = Z = \sum_{1\le i\le n} \bar{v}_i(\mathcal{F}). \tag{4.8}$$

However, Inequality (4.8) contradicts the fact that allocation $\mathcal{F}$ maximizes the social welfare. Hence, $\mathcal{F}$ guarantees at least 1/4 to all the agents.

Finally, let $\mathcal{R}$ be the randomized allocation obtained from $\mathcal{F}$ through (4.4). Then $v_i(\mathcal{R}) \ge v_i(\mathcal{F}) \ge \bar{v}_i(\mathcal{F}) \ge 1/4$ by Lemma 4.9. Thus, $\mathcal{R}$ is 1/4-MMS ex-ante. This completes the proof. □

We remark that though we constructed an ex-ante 1/4-MMS allocation, we have no guarantee on the ex-post fairness of our allocation. In the next section, our goal is to improve this allocation to also guarantee a fraction of maximin-share ex-post.

4.3 Ex-ante and Ex-post Guarantees

Unfortunately, the randomized allocation obtained by Theorem 4.1 has no ex-post fairness guarantee. The issue is that we use Theorem 4.13 to convert the fractional allocation into a randomized one. However, Theorem 4.13 only guarantees that the ex-post value of each agent is at least the value of her fractional allocation minus the value of the heaviest item which is partially (and not fully) allocated to her in the fractional allocation. However, currently, we have no upper bound on the value of the allocated items, and therefore, the ex-post value of an agent might be close to 0. To resolve this, we perform two improvements on our allocation.

First, we allocate valuable items beforehand to keep the value of the remaining items as small as possible. We start by using a simple and very practical fact that is frequently used in previous studies [7, 13, 44, 63]: allocating one item to one agent and removing them from the instance does not decrease the maximin-share value of the remaining agents for the remaining items.

Lemma 4.19. *Removing one item and one agent from the instance does not decrease the maximin-share value of the remaining agents for the remaining items.*

Given that our goal is to construct a randomized allocation which is 1/4-MMS ex-ante, by Lemma 4.19 we can assume without loss of generality that the value of each item to each agent is less than 1/4; otherwise, we can reduce the problem using Lemma 4.19. However, a combination of this assumption and Theorem 4.1 still gives no ex-post guarantee: the ex-ante guarantee obtained by Theorem 4.1 is 1/4-MMS and assuming that the value of each item to each agent is less than 1/4 implies no lower-bound better than 0 on the ex-post MMS guarantee. To improve the ex-post guarantee, we revisit the proof of Theorem 4.13 and show that for our setting, a stronger guarantee can be achieved using the matching method for converting a fractional allocation into a randomized one. Indeed, we show that we can find a fractional allocation with a special structure that makes the transformation step more efficient. These ideas together help us achieve a randomized allocation with 1/4-MMS guarantee ex-ante and 1/8-MMS guarantee ex-post.

Input: Instance $(\mathcal{N}, \mathcal{M}, \mathcal{V})$.
Output: Randomized allocation $\mathcal{R}$.

while there exists $g_j \in \mathcal{M}$ and $i \in \mathcal{N}$ s.t. $v_i(g_j) \geq 1/4$ **do** ▷ Step (1)
    $\mathcal{R}_i \leftarrow \{g_j\}$
    $\mathcal{M} \leftarrow \mathcal{M} \setminus \{g_j\}$
    $\mathcal{N} \leftarrow \mathcal{N} \setminus \{i\}$
Let $\bar{v}_i(\cdot) = \min(1/2, v_i(\cdot))$
Let Π be the set of all half-integral allocations of $\mathcal{M}$ to $\mathcal{N}$
Let $\mathcal{F} = \arg\max_{F \in \Pi} \sum_{i \in \mathcal{N}} \bar{v}_i(F_i)$ ▷ Step (2)
Let $\mathcal{R}$ be the randomized allocation obtained from $\mathcal{F}$ by Lemma 4.21. ▷ Step (3)
Return $\mathcal{R}$

Algorithm 8: `ExPostExAnteMMS`$(\mathcal{N}, \mathcal{M}, \mathcal{V})$

Theorem 4.2. *For any instance with* XOS *valuations, Algorithm 8 returns a randomized allocation that is* 1/4-MMS *ex-ante and* 1/8-MMS *ex-post.*

In the rest of this section, we prove Theorem 4.2. Algorithm 8 is as follows:

(1) While there exists an item g_j with value at least 1/4 to an agent i, allocate g_j to agent i and remove i and g_j respectively from $\mathcal{N}$ and $\mathcal{M}$.

(2) Assuming $\mathcal{N}'$ and $\mathcal{M}'$ are the set of the remaining agents and items respectively, let $\mathcal{F}$ be an optimal solution of the following linear program.

$$\begin{aligned}
\text{maximize} \quad & \sum_{i \in \mathcal{N}'} u_i \\
\text{subject to} \quad & \sum_{i \in \mathcal{N}'} f_{ij} = 1 && \text{for all } g_j \in \mathcal{M}' \\
& f_{ij} \in \{0, 1/2, 1\} && \text{for all } i \in \mathcal{N}' \text{ and } g_j \in \mathcal{M}' \\
& u_i = \min(\frac{1}{2}, \max_k \sum_j u_{i,k}(g_j) f_{ij}). && \text{for all } i \in \mathcal{N}' \qquad (4.9)
\end{aligned}$$

(3) Convert $\mathcal{F}$ into a randomized allocation using Lemma 4.21.

See Algorithm 8 for the pseudocode. Recall that by Lemma 4.19, after Step (1), the maximin-share value of the remaining agents for the remaining items is at least 1. For simplicity, we scale the valuations after the first step so that the MMS value of each remaining agent after the first step is exactly equal to 1. All agents i who are allocated an item g_j in Step (1), are also allocated g_j in the fractional allocation. Thus, 1/4-MMS is guaranteed for i in the final randomized allocation both ex-ante and ex-post.

An equivalent description for Step (2) is the following. For every agent $i \in \mathcal{N}'$, define $\bar{v}_i$ as follows:

$$\forall S \subseteq \mathcal{M}' \quad \bar{v}_i(S) = \min(1/2, v_i(S)).$$

Let $\bar{\mathcal{V}} = (\bar{v}_1, \ldots, \bar{v}_n)$ and return a half-integral allocation $\mathcal{F}$ that maximizes social welfare with respect to $\bar{\mathcal{V}}$, i.e., $\mathcal{F} = \arg\max_{A \in \Pi} \sum_{i \in \mathcal{N}'} \bar{v}_i(A_i)$ where Π is the set of all half-integral allocations of $\mathcal{M}'$ to $\mathcal{N}'$. The goal in Step (2) is to find a fractional allocation that is 1/4-MMS. However, we want this allocation to have a special structure that facilitates constructing the randomized allocation. Therefore, instead of directly choosing the allocation that maximizes social welfare, we consider $\bar{v}_i$ as the valuation function of agent i and return a half-integral allocation. First we prove that $\mathcal{F}$ is 1/4-MMS. Otherwise, let i^* be an agent that has a value less than 1/4 for her share. By Claim 4.18, we know that agent i^* can distribute all the items (that have remained after Step (1)) into $2n$ bundles each with value at least 1/2 to her. Here, we construct these $2n$ bundles more carefully.

Indeed, for every bundle in the optimal partitioning of agent i^*, we construct two bundles with a value of at least 1/2 as follows: we divide each item into two half-unit items and put each half-unit into one bundle. That way, for all items g_j, there are two bundles each of which contains one half of g_j.

Using the same deduction as we used in the proof of Theorem 4.2, we can say that since the number of remaining agents after Step (1) is n, the value of one agent for her bundle is less than 1/4, and the value of the rest of the agents for their bundles is at most 1/2, the social welfare of the allocation is less than $n/2$. Therefore, at least one of these $2n$ bundles, say B_k contributes less than 1/4 to social welfare. Now we take back these items from other agents and allocate them to agent i^*. The reallocation increases social welfare as shown in the proof of Theorem 4.1. Note that also in this new allocation for every agent i and item g_j, we have $f_{ij} \in \{0, 1/2, 1\}$ which is a contradiction with the choice of $\mathcal{F}$.

Observation 4.20. *Let $\mathcal{F}$ be the allocation after Step (2). Then, for every agent i we have $v_i(\mathcal{F}) \geq 1/4$. Furthermore, for every item g_j, we have $f_{ij} \in \{0, 1/2, 1\}$.*

Proof. Towards a contradiction assume $v_{i^*}(\mathcal{F}) < 1/4$ for some agent i^*. Let $B_1, B_2, \ldots, B_{2n}$ be the result of halving the bundles in the optimal partitioning of agent i^*, i.e., dividing each item into two half-unit items and putting each half-unit into one bundle. Let b_{kj} be the fraction of item g_j belonging to bundle B_k and let $\mathcal{F}'$ be the allocation (see Program 4.5) that defines the contribution of bundle B_k to allocation $\mathcal{F}$. Then $\sum_i f'_{ij} = \sum_i f_{ij} - b_{kj}$ for all j and $\sum_{1 \leq i \leq n} \bar{v}_i(\mathcal{F}) - \sum_{1 \leq i \leq n} \bar{v}_i(\mathcal{F}') = C^{\mathcal{F}}_{\bar{\mathcal{V}}}(B_k) < 1/4$. We now assign the items in B_k to agent i^*, i.e., we consider the fractional allocation $\mathcal{F}''$ equal to $\mathcal{F}'$, except that for agent i^*, we have

$$\forall_{1 \leq j \leq m} \qquad f''_{i^*j} = f'_{i^*j} + b_{kj}.$$

Since the value of bundle B_k to agent i^* is at least 1/2, we have $\bar{v}_{i^*}(\mathcal{F}'') - \bar{v}_{i^*}(\mathcal{F}') > 1/4$, and hence $\sum_{1 \leq i \leq n} \bar{v}_i(\mathcal{F}'') > \sum_{1 \leq i \leq n} \bar{v}_i(\mathcal{F})$. Since for every agent i and item g_j, we have $f''_{ij} \in \{0, 1/2, 1\}$ and hence a contradiction to the optimality of $\mathcal{F}$. □

Now, we show how to convert $\mathcal{F}$ into a randomized allocation. Recall that the result of Theorem 4.13 does not provide us with an ex-post guarantee better than 0. Here, we give a more accurate analysis to prove that the outcome of our algorithm is 1/8-MMS. Our construction is based on the Birkhoff—von Neumann theorem [16, 69]: Every fractional perfect matching can be written as a linear combination of integral perfect matchings. We adopt the construction to our setting and, in particular, exploit the fact that all f_{ij} are half-integral.

Lemma 4.21. *Assume that the valuations are additive and let $\mathcal{F}$ be a complete fractional allocation with $f_{ij} \in \{0, 1/2, 1\}$ for all i and j. Then there is a randomized allocation $\mathcal{R}$ with $D(\mathcal{R}) = \{\mathcal{A}^1, \mathcal{A}^2\}$, such that*

- *For every agent i we have $v_i(\mathcal{R}) = v_i(\mathcal{F})$.*
- *For every agent i we have*

$$\min\left\{v_i(\mathcal{A}_i^1), v_i(\mathcal{A}_i^2)\right\} \geq v_i(\mathcal{R}) - \frac{\max\{v_i(g_j) \mid f_{ij} = 1/2\}}{2}.$$

Proof. For each agent i, let $f_i = \sum_j f_{ij}$. Since $\sum_i f_i = m$, the number of agents with non-integral f_i is even. We pair the agents with non-integral f_i arbitrarily. For each pair, we create a new dummy item with a value of zero for all the agents and assign one-half of the dummy item to each agent in the pair. In this way, for every agent i, f_i becomes an integer. Therefore, for the rest of the proof we assume that for every agent i, f_i is an integer.

We now construct allocations $\mathcal{A}^1$ and $\mathcal{A}^2$ such that f_{ij} is equal to the fraction of allocations in $\mathcal{R}$ that allocate item g_j to i, i.e., if $f_{ij} = 1$ we allocate g_j to i in both allocations, if $f_{ij} = 0$, we allocate g_j to i in neither allocations, and if $f_{ij} = 1/2$ we allocate g_j to i in exactly one of the two allocations. For brevity, we define $\mathcal{M}_{1/2}$ and $\mathcal{N}_{1/2}$ as follows:

$$\mathcal{M}_{1/2} = \{g_j | \exists i : f_{ij} = 1/2\}.$$
$$\mathcal{N}_{1/2} = \{i | \exists g_j : f_{ij} = 1/2\}.$$

Consider a bipartite graph $G(X, Y)$ with parts X and Y as follows: for every item $g_j \in \mathcal{M}_{1/2}$ there is a vertex y_j in Y corresponding to item g_j. For every agent $i \in \mathcal{N}_{1/2}$, we have vertices $x_i^1, x_i^2, \ldots, x_i^{t_i}$ in X, where t_i is half the number of items g_j such that $f_{ij} = 1/2$, that is

$$t_i = \frac{\left|\{g_j | f_{ij} = 1/2\}\right|}{2}.$$

Also, we add the following edges to G. For every $i \in \mathcal{N}_{1/2}$, order the items g_j with $f_{ij} = 1/2$ in decreasing order of their value to agent i. Then we connect x_i^1 to the first two items, x_i^2 to the items with ranks three and four, and so on. In this way, all vertices in X and Y have degree two. Hence, G decomposes into vertex disjoint cycles. Each cycle decomposes into two matchings (note that since the graph is bipartite, all the cycles have even length), and thus G decomposes into two perfect matchings, say M^1 and M^2. We define allocations $\mathcal{A}^1$ and $\mathcal{A}^2$ as follows: for every item g_j, agent i, and $r \in \{1, 2\}$, we allocate item g_j to agent i in $\mathcal{A}^r$, if and only if either $f_{ij} = 1$, or $M^r(y_j) = x_i^k$ for some $1 \leq k \leq t_i$, where $M^r(y_j)$ refers to the vertex matched with y_j in M^r. Define $\mathcal{R}$ as a randomized allocation that selects $\mathcal{A}^1$ or $\mathcal{A}^2$, each with probability 1/2.

It is easy to check that for every agent i, $v_i(\mathcal{R}) = v_i(\mathcal{F})$. Here, we focus on the ex-post guarantee of $\mathcal{R}$. Fix an agent i and Let $g_1, g_2, \ldots, g_{2t_i}$ be the items half-owned by i in order of decreasing value for agent i. Then, by the way we construct M^1 and M^2, for all

$1 \leq \ell \leq t_i$, $g_{2\ell-1}$ and $g_{2\ell}$ are allocated to i in different allocations. Hence, the value of the i^{th} bundle in either allocation $\mathcal{A}^r$ satisfies

$$v_i(\mathcal{A}_i^r) \geq v_i(g_2) + v_i(g_4) + \ldots + v_i(g_{2t_i}) + \sum_{j:f_{ij}=1} v_i(g_j).$$

We can now bound $v_i(\mathcal{F}) - v_i(\mathcal{A}_i^r)$ from above.

$$\begin{aligned} v_i(\mathcal{F}) - v_i(\mathcal{A}_i^r) &\leq \frac{1}{2}\left(\sum_{1\leq \ell \leq 2t_i} v_i(g_\ell)\right) - \sum_{1\leq \ell \leq t_i} v_i(g_{2\ell}) \\ &= \frac{v_i(g_1)}{2} + \sum_{1\leq \ell < t_i}\left(\frac{v_i(g_{2\ell}) + v_i(g_{2\ell+1})}{2} - v_i(g_{2\ell})\right) + \frac{v_i(g_{2t_i})}{2} - v_i(g_{2t_i}) \\ &\leq \frac{v_i(g_1)}{2}. \end{aligned}$$

□

Now we are ready to prove Theorem 4.2.

Theorem 4.2. *For any instance with* XOS *valuations, Algorithm 8 returns a randomized allocation that is* 1/4-MMS *ex-ante and* 1/8-MMS *ex-post.*

Proof. The ex-ante guarantee follows from Observation 4.20 and Lemma 4.4.

Let $\mathcal{A}^1$ and $\mathcal{A}^2$ be the integral allocations obtained by Lemma 4.21. Consider any agent i, and let $u_{i,i'}$ be such that $v_i(\mathcal{R}) = \sum_j f_{ij} u_{i,i'}(g_j)$. Then, by Lemma 4.21 for $r \in \{1, 2\}$ we have

$$u_{i,i'}(\mathcal{A}_i^r) \geq u_{i,i'}(\mathcal{R}_i^\ell) - \frac{\max\{u_{i,i'}(g_j) \mid f_{ij} = 1/2\}}{2}.$$

Since by Lemma 4.19 we know the value of each item for each agent is less than 1/4, we have

$$v_i(\mathcal{A}_i^\ell) \geq v_i(\mathcal{R}_i) - \frac{\max_j v_i(g_j)}{2} > \frac{1}{4} - \frac{1}{8} = \frac{1}{8}.$$

Hence, the ex-post guarantee holds as well. □

4.4 3/13-MMS Allocation

In this section, we improve the best approximation guarantee of MMS for deterministic allocations in the fractionally subadditive setting. We show that a factor $3/13 \approx 0.230769$ of the maximin-share of every agent is possible. Before this work, the best approximation guarantee for maximin-share in the XOS setting was 0.2192235-MMS [63].

Our algorithm for improving the ex-post guarantee is based on our previous algorithms plus two additional steps and a more in-depth analysis. In this algorithm, before finding the allocation that maximizes social welfare, we strengthen our upper bound on the value of items. For this, we add two more steps to our algorithm in which we satisfy some of the agents with two items and three items. In contrast to the first step (i.e., allocating single items to agents), these steps might decrease the maximin-share value of the remaining agents for the remaining items. Let $t = 6/13$. The goal is to find a $t/2$-MMS allocation. Our allocation algorithm is as follows:

Input: Instance $(\mathcal{N}, \mathcal{M}, \mathcal{V})$.
Output: Allocation $\mathcal{A}$.

Let $t = 6/13$
while there exists $g_j \in \mathcal{M}$ and $i \in \mathcal{N}$ s.t. $v_i(g_j) \geq t/2$ **do** ▷ Step 1
    $\mathcal{A}_i \leftarrow \{g_j\}$
    $\mathcal{M} \leftarrow \mathcal{M} \setminus \{g_j\}$
    $\mathcal{N} \leftarrow \mathcal{N} \setminus \{i\}$
while there exists $g_j, g_k \in \mathcal{M}$ and $i \in \mathcal{N}$ s.t. $v_i(\{g_j, g_k\}) \geq t/2$ **do** ▷ Step 2
    $\mathcal{A}_i \leftarrow \{g_j, g_k\}$
    $\mathcal{M} \leftarrow \mathcal{M} \setminus \{g_j, g_k\}$
    $\mathcal{N} \leftarrow \mathcal{N} \setminus \{i\}$
while there exists $g_j, g_k, g_s \in \mathcal{M}$ and $i \in \mathcal{N}$ s.t. $v_i(\{g_j, g_k, g_s\}) \geq t/2$ **do** ▷ Step 3
    $\mathcal{A}_i \leftarrow \{g_j, g_k, g_s\}$
    $\mathcal{M} \leftarrow \mathcal{M} \setminus \{g_j, g_k, g_s\}$
    $\mathcal{N} \leftarrow \mathcal{N} \setminus \{i\}$
Let $\mathcal{N}' = \mathcal{N}$, $\mathcal{M}' = \mathcal{M}$ and $\bar{v}_i(\cdot) = \min(t, v_i(\cdot))$
Let Π be the set of all allocations of $\mathcal{M}'$ to $\mathcal{N}'$
Let $\mathcal{A} = \arg\max_{A \in \Pi} \sum_{i \in \mathcal{N}'} \bar{v}_i(A_i)$ ▷ Step 4
Return $\mathcal{A}$

Algorithm 9: `approxMMS`$(\mathcal{N}, \mathcal{M}, \mathcal{V})$

(1) While there exists an item g_j with value at least $t/2$ to an agent i, allocate g_j to agent i and remove i and g_j respectively from $\mathcal{N}$ and $\mathcal{M}$.

(2) While there exists a pair of items g_j, g_k with total value of at least $t/2$ to some agent i, allocate $\{g_j, g_k\}$ to agent i, remove both goods from $\mathcal{M}$, and remove agent i from $\mathcal{N}$.

(3) While there exists a triple of items g_j, g_k, g_s with total value of at least $t/2$ to some agent i, allocate $\{g_j, g_k, g_s\}$ to agent i, remove all three goods from $\mathcal{M}$, and remove agent i from $\mathcal{N}$.

(4) For the remaining agents $\mathcal{N}'$ and items $\mathcal{M}'$, proceed as follows: for every agent i, define $\bar{v}_i$ as follows:

$$\forall S \subseteq \mathcal{M} \quad \bar{v}_i(S) = \min(t, v_i(S)).$$

Let $\bar{\mathcal{V}} = (\bar{v}_1, \ldots, \bar{v}_n)$ and return an allocation $\mathcal{A}$ that maximizes social welfare with respect to $\bar{v}$, i.e., $\mathcal{A} = \arg\max_{A \in \Pi} \sum_{i \in \mathcal{N}'} \bar{v}_i(A_i)$ where Π is the set of all allocations of $\mathcal{M}'$ to $\mathcal{N}'$.

In the rest of this section, we analyze the above algorithm. See Algorithm 9 for the pseudocode. By Lemma 4.19, after Step (1), the MMS value of all the agents is at least 1. Let n be the number of the remaining agents after Step (1). We denote by n_1 and n_2, the number of agents that are satisfied in Steps (2) and (3) respectively and let $n' = n - n_1 - n_2 = |\mathcal{N}'|$ be the number of remaining agents after Step (3). In contrast to the first step, Step (2) and (3) might decrease the maximin-share value of the remaining agents for the remaining items. However, we prove that the remaining items satisfy certain special structural properties.

Observation 4.22. *Since no item can satisfy any remaining agent after Step (1), for every agent i and every item g_j, we have $v_i(g_j) < t/2$.*

Also, by the method that we allocate the items in Step (3), after this step the following observation holds.

Observation 4.23. *Since after Step (3), no triple of items can satisfy an agent, for every different items g_j, g_k, g_s and every agent i we have $v_i(\{g_j, g_k, g_s\}) < t/2$.*

Note that since the valuations are XOS, Observation 4.23 implies no upper bound better than $t/2$ on the value of a single item to an agent. For example, consider the following extreme scenario: for a small constant $\varepsilon > 0$, the value of every non-empty subset of items to agent i is equal to $t/2 - \varepsilon$. It is easy to check that this valuation function is XOS. For this case, the value of every triple of items is also equal to $t/2 - \varepsilon$, but this implies no upper bound better than $t/2$ on the value of a single item.

Lemma 4.24. *Fix a remaining agent i and consider the n bundles with value at least 1 in an* MMS *partition of agent i after Step (1). Put these bundles into 4 different sets $B_0, B_1, B_2, B_{\geq 3}$, where for $0 \leq \ell \leq 2$, set B_ℓ contains bundles that lose exactly ℓ items in Steps (2) and (3), and $B_{\geq 3}$ contains bundles that lose at least three items in these steps. After Step (3), the following inequality holds:*

$$n' \leq |B_0| + \frac{2}{3}|B_1| + \frac{1}{3}|B_2|.$$

Proof. Since each satisfied agent in step k receives k items, we have:

$$2n_1 + 3n_2 \geq |B_1| + 2|B_2| + 3|B_{\geq 3}|.$$

Thus,

$$n_1 + n_2 \geq \frac{2}{3}n_1 + n_2 \geq \frac{1}{3}|B_1| + \frac{2}{3}|B_2| + |B_{\geq 3}|, \tag{4.10}$$

and therefore,

$$\begin{aligned} n' &= n - n_1 - n_2 \\ &\leq n - \frac{1}{3}|B_1| - \frac{2}{3}|B_2| - |B_{\geq 3}| && \text{(Inequality 4.10)} \\ &= |B_0| + \frac{2}{3}|B_1| + \frac{1}{3}|B_2|. && (n = |B_0| + |B_1| + |B_2| + |B_{\geq 3}|) \end{aligned}$$

□

Finally, in Step (4), we find the integral allocation $\mathcal{A}$ that maximizes social welfare with respect to $\bar{v}$ for the remaining agents. Let

$$Z = \sum_{i \in \mathcal{N}'} \bar{v}_i(\mathcal{A}_i).$$

Since for each remaining agent i, $\bar{v}_i(\mathcal{A}_i)$ is upper-bounded by t, we have $Z \leq n't$. If for every agent i, $v_i(\mathcal{A}_i) \geq t/2$ holds, then $\mathcal{A}$ is $t/2$-MMS, and we are done. Therefore, for the rest of this section, assume that for an agent i^*, we have $v_{i^*}(\mathcal{A}_{i^*}) < t/2$.

Lemma 4.25. *For all sets* $S \subseteq M$, $C^{\mathcal{A}}_{\bar{\mathcal{V}}}(S) \geq \bar{v}_{i^*}(S) - \bar{v}_{i^*}(\mathcal{A}_{i^*})$.

Proof. Let allocation $\mathcal{A}'$ be as following. For all agents i, $\mathcal{A}'_i = \mathcal{A}_i \setminus S$. Basically, $\mathcal{A}'$ is allocation $\mathcal{A}$ after removing all the items in S from the bundles they belong to. We have

$$\sum_{i \in \mathcal{N}} \bar{v}_i(\mathcal{A}'_i) = \sum_{i \in \mathcal{N}} \bar{v}_i(\mathcal{A}_i) - C^{\mathcal{A}}_{\bar{\mathcal{V}}}(S). \tag{4.11}$$

Now let $\mathcal{A}''$ be allocation $\mathcal{A}'$ after allocating S to agent i^*. I.e., for all agents $i \neq i^*$, $\mathcal{A}''_i = \mathcal{A}'_i$ and $\mathcal{A}''_{i^*} = \mathcal{A}'_{i^*} \cup S$. We have

$$\begin{aligned}
\sum_{i \in \mathcal{N}} \bar{v}_i(\mathcal{A}_i) &\geq \sum_{i \in \mathcal{N}} \bar{v}_i(\mathcal{A}''_i) && (\mathcal{A} = \arg\max_{A \in \Pi} \textstyle\sum_{i \in \mathcal{N}'} \bar{v}_i(A_i)) \\
&= \sum_{i \in \mathcal{N} \setminus \{i^*\}} \bar{v}_i(\mathcal{A}'_i) + \bar{v}_{i^*}(\mathcal{A}'_{i^*} \cup S) \\
&= \left(\sum_{i \in \mathcal{N}} \bar{v}_i(\mathcal{A}_i) - C^{\mathcal{A}}_{\bar{\mathcal{V}}}(S) - \bar{v}_{i^*}(\mathcal{A}'_{i^*}) \right) + \bar{v}_{i^*}(\mathcal{A}'_{i^*} \cup S) && \text{(Inequality (4.11))} \\
&\geq \sum_{i \in \mathcal{N}} \bar{v}_i(\mathcal{A}_i) - C^{\mathcal{A}}_{\bar{\mathcal{V}}}(S) - \bar{v}_{i^*}(\mathcal{A}'_{i^*}) + \bar{v}_{i^*}(S). && (\bar{v}_{i^*}(\mathcal{A}'_{i^*} \cup S) \geq \bar{v}_{i^*}(S))
\end{aligned}$$

Therefore, $C^{\mathcal{A}}_{\bar{\mathcal{V}}}(S) \geq \bar{v}_{i^*}(S) - \bar{v}_{i^*}(\mathcal{A}_{i^*})$. □

Let B_0, B_1, and B_2 be the sets defined for agent i^* in Lemma 4.24. In Lemmas 4.26, 4.27 and 4.28, we give lower bounds on the contribution of the bundles in B_0, B_1 and B_2 to $\mathcal{A}$ respectively.

Lemma 4.26. *After Step (3), for all bundles* $X \in B_0$*, there exists a partition of* X *into* X_1 *and* X_2 *such that* $C^{\mathcal{A}}_{\bar{\mathcal{V}}}(X_1) + C^{\mathcal{A}}_{\bar{\mathcal{V}}}(X_2) \geq t$.

Proof. The idea is to partition the set X into two bundles X_1 and X_2 each with value at least t to agent i^*. Then using Lemma 4.25, we prove the contribution of each of these bundles to $\mathcal{A}$ is at least $t/2$ and thus the total contribution is at least t.

For a fixed bundle $X \in B_0$, let j be such that $u_{i^*,j}(X) = v_{i^*}(X) \geq 1$. Let g_1 and g_2 be two most valuable items in X with respect to $u_{i^*,j}$, i.e., for all items $g \in X \setminus \{g_1, g_2\}$, $u_{i^*,j}(g_1) \geq u_{i^*,j}(g_2) \geq u_{i^*,j}(g)$. Let X_1 be a minimal subset of X such that $\{g_1, g_2\} \subset X_1$ and $u_{i^*,j}(X_1) \geq t$. Let $X_2 = X \setminus X_1$. Since X_1 is minimal, for all $g \in X_1$, $u_{i^*,j}(X_1 \setminus \{g\}) < t$. Also, by Observation 4.23, for all $g \in X_1 \setminus \{g_1, g_2\}$, $u_{i^*,j}(\{g_1, g_2, g\}) \leq v_{i^*}(\{g_1, g_2, g\}) < t/2$ and thus, $u_{i^*,j}(g) < t/6$. Therefore, for all $g \in X_1 \setminus \{g_1, g_2\}$,

$$\begin{aligned}
u_{i^*,j}(X_2) &\geq 1 - u_{i^*,j}(X_1) && (u_{i^*,j}(X_1 \cup X_2) \geq 1) \\
&= 1 - (u_{i^*,j}(X_1 \setminus \{g\}) + u_{i^*,j}(g)) && \text{(by additivity of } u_{i^*,j}) \\
&> 1 - \frac{7}{6}t \\
&= t. && (t = 6/13)
\end{aligned}$$

Hence, we have $v_{i^*}(X_1) \geq u_{i^*,j}(X_1) \geq t$ and $v_{i^*}(X_2) \geq u_{i^*,j}(X_2) \geq t$. Now by Lemma 4.25, we have

$$\begin{aligned}
C^{\mathcal{A}}_{\bar{\mathcal{V}}}(X_1) + C^{\mathcal{A}}_{\bar{\mathcal{V}}}(X_2) &\geq (\bar{v}_{i^*}(X_1) - \bar{v}_{i^*}(\mathcal{A}_{i^*})) + (\bar{v}_{i^*}(X_2) - \bar{v}_{i^*}(\mathcal{A}_{i^*})) \\
&> 2(t - \frac{1}{2}t) = t.
\end{aligned}$$

□

Lemma 4.27. *After Step (3), for all bundles $X \in B_1$, there exists a partition of X into X_1 and X_2 such that $C^{\mathcal{A}}_{\bar{\mathcal{V}}}(X_1) + C^{\mathcal{A}}_{\bar{\mathcal{V}}}(X_2) \geq \frac{2}{3}t$.*

Proof. Fix a bundle $X \in B_1$. By Lemma 4.19, the MMS value of agent i^* is at least 1 after Step (1). By Observation 4.22, $v_{i^*}(g) < t/2$ for all remaining items g after Step (1). Since X is a bundle in an MMS partition of agent i^* after Step (1) and after the removal of one item g, we have

$$v_{i^*}(X) > 1 - t/2. \tag{4.12}$$

Let j be such that $u_{i^*,j}(X) = v_{i^*}(X)$. Let g_1 and g_2 be two most valuable items in X with respect to $u_{i^*,j}$, i.e., for all items $g \in X \setminus \{g_1, g_2\}$, $u_{i^*,j}(g_1) \geq u_{i^*,j}(g_2) \geq u_{i^*,j}(g)$. Let X_1 be a minimal subset of X such that $\{g_1, g_2\} \subset X_1$ and $u_{i^*,j}(X_1) \geq 2t/3$. Let $X_2 = X \setminus X_1$. Since X_1 is minimal, for all $g \in X_1$, $u_{i^*,j}(X_1 \setminus \{g\}) < 2t/3$. Also, by Observation 4.23, for all $g \in X_1 \setminus \{g_1, g_2\}$, $u_{i^*,j}(\{g_1, g_2, g\}) \leq v_{i^*}(\{g_1, g_2, g\}) < t/2$ and thus, $u_{i^*,j}(g) < t/6$. Therefore,

$$\begin{aligned} u_{i^*,j}(X_1) &= u_{i^*,j}(X_1 \setminus \{g\}) + u_{i^*,j}(g) && \text{(by additivity of } u_{i^*,j}) \\ &< \frac{2}{3}t + \frac{1}{6}t = \frac{5}{6}t. \end{aligned}$$

Therefore, for all $g \in X_1 \setminus \{g_1, g_2\}$, we have

$$\begin{aligned} C^{\mathcal{A}}_{\bar{\mathcal{V}}}(X_1) + C^{\mathcal{A}}_{\bar{\mathcal{V}}}(X_2) &\geq (\bar{v}_{i^*}(X_1) - \bar{v}_{i^*}(\mathcal{A}_{i^*})) + (\bar{v}_{i^*}(X_2) - \bar{v}_{i^*}(\mathcal{A}_{i^*})) && \text{(Lemma 4.25)} \\ &= (\min(t, v_{i^*}(X_1)) - \bar{v}_{i^*}(\mathcal{A}_{i^*})) + (\min(t, v_{i^*}(X_2)) - \bar{v}_{i^*}(\mathcal{A}_{i^*})) \\ &> \left(\min(t, u_{i^*,j}(X_1)) - \frac{1}{2}t\right) + \left(\min(t, u_{i^*,j}(X_2)) - \frac{1}{2}t\right) \\ &\geq u_{i^*,j}(X_1) + \min(t, 1 - \frac{1}{2}t - u_{i^*,j}(X_1)) - t && (u_{i^*,j}(X_1) < 5t/6) \\ &\geq \min(u_{i^*,j}(X_1), 1 - \frac{3}{2}t) \\ &\geq \frac{2}{3}t. \end{aligned}$$

□

Lemma 4.28. *After Step (3), for all bundles $X \in B_2$, $C^{\mathcal{A}}_{\bar{\mathcal{V}}}(X) \geq \frac{1}{2}t$.*

Proof. Fix a bundle $X \in B_2$. By Lemma 4.19, the MMS value of agent i^* is at least 1 after Step (1). By Observation 4.22, $v_{i^*}(g) < t/2$ for all remaining items g after Step (1). Since X is a bundle in an MMS partition of agent i^* after Step (1) and after the removal of two items like g, we have $v_{i^*}(X) > 1 - t > t$. Therefore, $\bar{v}_{i^*}(X) = \min(t, v_{i^*}(X)) = t$. Now by Lemma 4.25,

$$C^{\mathcal{A}}_{\bar{\mathcal{V}}}(X) \geq \bar{v}_{i^*}(X) - \bar{v}_{i^*}(\mathcal{A}_i) > t - \frac{1}{2}t = \frac{1}{2}t.$$

□

Theorem 4.3. *For any instance with* XOS *valuations, Algorithm 9 returns a* 3/13-MMS *allocation.*

Proof. Let $\mathcal{A}$ be the output of Algorithm 9. Towards a contradiction, assume for agent i^*, $v_{i^*}(\mathcal{A}_{i^*}) < 3/13 = t/2$. For all agents i which are removed during the first three steps, we have $v_i(\mathcal{A}_i) \geq t/2 = 3/13$. Therefore, $i^* \in \mathcal{N}'$. For all $X \in B_0$, let X_1 and X_2 be as defined in Lemmas 4.26 and 4.27. We have

$$\begin{aligned}
t(n' - \frac{1}{2}) &> \sum_{i \in \mathcal{N}'} \bar{v}_i(\mathcal{A}_i) && \text{(for all } i \in \mathcal{N}', \bar{v}_i(\mathcal{A}_i) \leq t \text{ and } \bar{v}_{i^*}(\mathcal{A}_{i^*}) < t/2) \\
&\geq \sum_{X \in B_0} \left(C^{\mathcal{A}}_{\mathcal{V}}(X_1) + C^{\mathcal{A}}_{\mathcal{V}}(X_2)\right) + \sum_{X \in B_1} \left(C^{\mathcal{A}}_{\mathcal{V}}(X_1) + C^{\mathcal{A}}_{\mathcal{V}}(X_2)\right) + \sum_{X \in B_2} C^{\mathcal{A}}_{\mathcal{V}}(X) && \text{(Lemma 4.10)} \\
&\geq t|B_0| + \frac{2}{3}t|B_1| + \frac{1}{2}t|B_2| && \text{(Lemmas 4.26, 4.27 and 4.28)} \\
&\geq tn', && \text{(Lemma 4.24)}
\end{aligned}$$

which is a contradiction. Therefore, such an agent i^* does not exist and $\mathcal{A}$ is a 3/13-MMS allocation. □

CHAPTER 5

Ordinal MMS Approximation

In the ordinal approximation, the goal is to show the existence of 1-out-of-d MMS allocations (for the smallest possible $d > n$). A series of works led to the state-of-the-art factor of $d = \lfloor 3n/2 \rfloor$ for additive valuations [47]. We show that 1-out-of-$4\lceil n/3 \rceil$ MMS allocations always exist when agents have additive valuation functions, thereby improving the state-of-the-art of ordinal MMS approximation.

While maybe counter-intuitive at first sight, there is no meaningful relation known between the multiplicative approximations and ordinal approximations of MMS. Consider the following examples.

(1) $m = n$ and $v_i(g) = 1$ for all $i \in \mathcal{N}$ and $g \in \mathcal{M}$. Then for all $i \in \mathcal{N}$, $\mathrm{MMS}_i^n(\mathcal{M}) = 1$ while $\mathrm{MMS}_i^d(\mathcal{M}) = 0$ for all $d > n$. Therefore, any allocations is a 1-out-of-$(n+1)$ MMS allocation, while it might not guarantee α-MMS property for any $\alpha > 0$.

(2) $m = 2n - 1$ and $v_i(g) = 1$ for all $i \in \mathcal{N}$ and $g \in \mathcal{M}$. Then $\mathrm{MMS}_i^n(\mathcal{M}) = \mathrm{MMS}_i^{2n-1}(\mathcal{M}) = 1$ for all $i \in \mathcal{N}$. Therefore, any allocation that satisfies 1-out-of-$(2n-1)$ MMS also guarantees MMS.

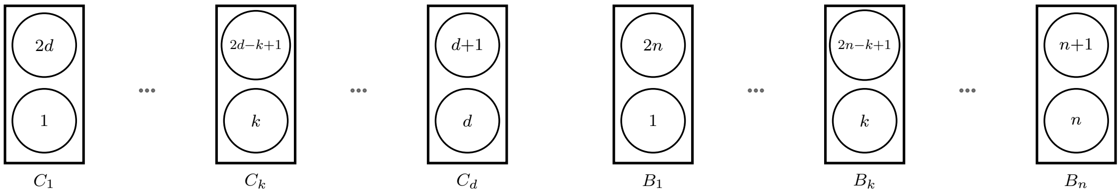
This also explains the discrepancy known between the previously best known α-MMS guarantee ($\alpha = 3/4$), and 1-out-of-(n/β) MMS guarantee ($\beta = 2/3$). It goes without saying that the $(3/4 + 3/3836)$-MMS result in Chapter 3 does not imply the existence of 1-out-of-$4\lceil n/3 \rceil$ MMS allocations, which we prove in this chapter—nor does the latter imply the former.

5.1 Notations and Tools

In this chapter, we assume all agents have additive valuation functions. We prove the existence of 1-out-of-$(4n/3)$ MMS assuming that n is a multiple of 3. This implies the existence of 1-out-of-$4\lceil n/3 \rceil$ MMS for all n. When n is not a multiple of 3, we can copy one of the agents 1 or 2 times (depending on $n \bmod 3$) so that the new instance has $n' := 3\lceil n/3 \rceil$ agents. Since we prove the existence of 1-out-of-$(4n'/3)$ MMS for the new instance, we prove the existence of an allocation that gives all the agents i in the original instance their $\mathrm{MMS}_i^{4n'/3}(\mathcal{M}) = \mathrm{MMS}_i^{4\lceil n/3 \rceil}(\mathcal{M})$ value. Hence, the existence of 1-out-of-$4\lceil n/3 \rceil$ MMS allocations follows. Recall that for a given instance $\mathcal{I}$ and integer d, for each agent i, $P^{(i)} = (P_1^{(i)}, \ldots, P_d^{(i)})$ is a d-MMS partition of agent i.

Similar to Chapter 3 where the goal was to find an α-MMS allocation, when the goal is to find a 1-out-of-d MMS allocation, it is without loss of generality to assume the instance is ordered and d-normalized. We prove it formally in Lemma 5.1.

Lemma 5.1. *For any $d \in \mathcal{N}$, if 1-out-of-d MMS allocations exist for d-normalized ordered instances, then 1-out-of-d MMS allocations exist for all instances.*



(a) For all $k \in [d]$, $C_k := \{k, 2d-k+1\}$. **(b)** For all $k \in [n]$, $B_k := \{k, 2n-k+1\}$.

Figure 5.1: Bag Initialization

Proof. Let $\mathcal{I}$ be an arbitrary instance. We create a d-normalized ordered instance $\mathcal{I}'' = (\mathcal{N}, \mathcal{M}, \mathcal{V}'')$ such that from any 1-out-of-d MMS allocation for $\mathcal{I}''$, one can obtain a 1-out-of-d MMS allocation for the original instance $\mathcal{I}$.

First of all, we can ignore all agents i with $\text{MMS}_i^d = 0$ since no good needs to be allocated to them. For all $i \in \mathcal{N}$ and $g \in \mathcal{M}$, we define $v'_{i,g} = v_i(g)/v_i(P_j^{(i)})$ where j is such that $g \in P_j^{(i)}$. Now for all $i \in \mathcal{N}$, let $v'_i : 2^{\mathcal{M}} \to \mathbb{R}_{\geq 0}$ be defined as an additive function such that $v'_i(S) = \sum_{g \in S} v'_{i,g}$. Note that $v'_{i,g} \leq v_i(g)/\text{MMS}_i^d(\mathcal{M})$ for all $g \in \mathcal{M}$ and thus,

$$v_i(S) \geq v'_i(S) \cdot \text{MMS}_i^d(\mathcal{M}). \tag{5.1}$$

Since $v'_i(P_j^{(i)}) = 1$ for all $i \in \mathcal{N}$ and $j \in [d]$, $\mathcal{I}' = (\mathcal{N}, \mathcal{M}, \mathcal{V}')$ is a d-normalized instance. If a 1-out-of-d MMS allocation exists for $\mathcal{I}$, let X be one such allocation. By Inequality (5.1), $v_i(X_i) \geq v'_i(X_i) \cdot \text{MMS}_i^d(\mathcal{M}) \geq \text{MMS}_i^d(\mathcal{M})$. Thus, every allocation that is 1-out-of-d MMS for $\mathcal{I}'$ is 1-out-of-d MMS for $\mathcal{I}$ as well. For all agents i and $g \in [m]$, let $v''_{i,g}$ be the g^{th} number in the multi-set of $\{v_i(1), \dots, v_i(m)\}$. Let $v''_i : 2^{\mathcal{M}} \to \mathbb{R}_{\geq 0}$ be defined as an additive function such that $v''_i(S) = \sum_{g \in S} v''_{i,g}$. Let $\mathcal{I}'' = (\mathcal{N}, \mathcal{M}, \mathcal{V}'')$. Note that $\mathcal{I}''$ is ordered and d-normalized. By Corollary 2.14, from any 1-out-of-d MMS allocation in $\mathcal{I}''$, one can obtain a 1-out-of-d MMS allocation in $\mathcal{I}'$ and as already shown before, it gives a 1-out-of-d MMS allocation for $\mathcal{I}$. ☐

Proposition 5.2. *Given a d-normalized instance for all $i \in \mathcal{N}$ and $k \in [d]$, we have*

(1) $v_i(P_k^{(i)}) = 1$, *and*

(2) $v_i(\mathcal{M}) = d$.

We note that it is without loss of generality to assume $m \geq 2d$. Otherwise, we can add $2d - m$ dummy goods with a value of 0 for all the agents. The normalized and ordered properties of the instance would be preserved. For the rest of this chapter, we assume the instance is ordered and d-normalized and $m \geq 2d$.

Consider the bag setting with d bags as follow.

$$C_k := \{k, 2d - k + 1\} \text{ for } k \in [d] \tag{5.2}$$

See Figure 5.1(a) for more intuition. Next, we show some important properties of the values of the goods in C_k's.

Proposition 5.3. *For all agents $i \in \mathcal{N}$, we have*

(1) $v_i(1) \leq 1$,

(2) $v_i(C_d) \leq 1$, *and*

(3) $v_i(d+1) \leq \frac{1}{2}$.

Proof. For the first part, fix an agent i. Let $1 \in P_1^{(i)}$. By Proposition 5.2, $v_i(1) \leq v_i(P_1^{(i)}) = 1$.

For the second part, by the pigeonhole principle, there exists a bundle $P_k^{(i)}$ and two goods $j, j' \in \{1, 2, \ldots, d+1\}$ such that $\{j, j'\} \subseteq P_k^{(i)}$. Without loss of generality, assume $j < j'$. We have

$$\begin{aligned} v_i(C_d) &= v_i(d) + v_i(d+1) && (C_d = \{d, d+1\}) \\ &\leq v_i(j) + v_i(j') && (j \leq d \text{ and } j' \leq d+1) \\ &\leq v_i(P_k^{(i)}) = 1. && (\{j, j'\} \in P_k^{(i)}) \end{aligned}$$

For the third part, we have

$$1 \geq v_i(C_d) = v_i(d) + v_i(d+1) \geq 2v_i(d+1).$$

Thus, $v_i(d+1) \leq \frac{1}{2}$. □

Lemma 5.4. *For all $i \in \mathcal{N}$ and $k \in [d]$, $\sum_{j=k}^{d} v_i(C_j) \leq d - k + 1$.*

Proof. For the sake of contradiction, assume the claim does not hold for some agent i and let $\ell \geq 1$ be the largest index for which we have $\sum_{j=\ell}^{d} v_i(C_j) > d - \ell + 1$. Proposition 5.3((2)) implies that $\ell < d$. We have

$$\begin{aligned} v_i(\ell) + v_i(2d - \ell + 1) &= v_i(C_\ell) \\ &= \sum_{j=\ell}^{d} v_i(C_j) - \sum_{j=\ell+1}^{d} v_i(C_j) \\ &> (d - \ell + 1) - (d - (\ell + 1) + 1) \\ &\qquad\qquad (\textstyle\sum_{j=k}^{d} v_i(C_j) \leq d - k + 1 \text{ for } k > \ell) \\ &= 1. \end{aligned}$$

For all $j, j' < \ell$, $v_i(j) + v_i(j') \geq v_i(\ell) + v_i(2d - \ell + 1) > 1$. Therefore, j and j' cannot be in the same bundle in any d-MMS partition of i. For $j < \ell$, let $j \in P_j^{(i)}$. For all $j < \ell$ and $\ell \leq j' \leq 2d - \ell + 1$,

$$\begin{aligned} v_i(j) + v_i(j') &\geq v_i(\ell) + v_i(2d - \ell + 1) \\ &= v_i(C_\ell) > 1. \end{aligned}$$

Therefore, $j' \notin P_j$. Also, since $\sum_{j=\ell}^{d} v_i(C_j) > d - \ell + 1$, there are at least $t \geq d - \ell + 2$ different bundles $Q_1, \ldots, Q_t$ in P such that $Q_j \cap \{\ell, \ldots, 2d - \ell + 1\} \neq \emptyset$. It is a contradiction since these $t \geq d - \ell + 2$ bundles must be different from $P_1^{(i)}, \ldots P_{\ell-1}^{(i)}$. □

Input: Ordered $4\lceil n/3 \rceil$-normalized instance $\mathcal{I} = (\mathcal{N}, \mathcal{M}, \mathcal{V})$.
Output: Allocation $\hat{B} = (\hat{B}_1, \ldots, \hat{B}_n)$.

for $k \in [n]$ **do** ▷ Initialization
  $B_k = \{k, 2n - k + 1\}$
$j \leftarrow 2n + 1$
for $k \in [n]$ **do** ▷ Bag-filling
  while $\nexists i \in \mathcal{N}$ s.t. $v_i(B_k) \geq 1$ **do**
    $B_k \leftarrow B_k \cup \{j\}$
    $j \leftarrow j + 1$
    if $j > m$ **then**
      Terminate
  Let $i \in \mathcal{N}$ s.t. $v_i(B_k) \geq 1$
  $\hat{B}_i \leftarrow B_k$
  $\mathcal{N} \leftarrow \mathcal{N} \setminus \{i\}$
$\hat{B}_n \leftarrow \hat{B}_n \cup (\mathcal{M} \setminus [j])$
return $\hat{B}$

Algorithm 10: 1-out-of-$4\lceil n/3 \rceil$ MMS

5.2 Technical Overview

For α-MMS problem, the algorithms for $\alpha \geq 3/4$ [42, 44] and Algorithm 7 utilize the two-phase approach: *valid reductions* and *bag-filling*. In a valid reduction, the instance is reduced by removing an agent a and a subset of goods S such that $v_a(S) \geq \alpha$, and the MMS values of the remaining agents do not decrease. The valid reduction phase is crucial for the bag-filling to work in the analysis of these algorithms. However, it is not clear how to define valid reductions in the case of 1-out-of-d MMS because d is not the same as the number of agents n. Therefore, we only use (a variation of) bag-filling in our algorithm, which makes its analysis quite involved and entirely different from the α-MMS algorithms.

Our algorithm is very simple, described in Algorithm 10. Given an ordered d-normalized instance, we initialize n bags (one for each agent) with the first $2n$ (highest valued) goods as follows.

$$B_k := \{k, 2n - k + 1\} \text{ for } k \in [n]. \tag{5.3}$$

See Figure 5.1(b) for a better intuition. Then, we do a slight variation of bag-filling. That is, in each round j, we keep adding goods of decreasing values to the bag B_j until some agent with no assigned bag values it at least 1 (recall that 1-out-of-d MMS value of each agent is 1 in a d-normalized instance). Then, we allocate it to an arbitrary such agent. We note that in contrast to [42] and Section 3.2, here we do not add arbitrary goods to arbitrary bags, but we add the goods in the decreasing order of their values to the bags in the increasing order order of their index. For the rest of the chapter, we simply use "bag-filling" to refer to this procedure.

To prove that the output of Algorithm 10 is 1-out-of-d MMS, it is sufficient to prove that we never run out of goods in any round or, equivalently, each agent receives a bag in some round. Towards contradiction, assume that agent i^* does not receive a bag, and the algorithm terminates. It can be easily argued that agent i^*'s value for at least one of the initial bags $\{B_1, \ldots, B_n\}$ must be strictly less than 1. Let ℓ^* be the smallest such that $v_{i^*}(B_{\ell^*}) < 1$. We consider two cases based on the value of $v_{i^*}(2n - \ell^*)$. In Section 5.3.1, we reach a contradiction assuming $v_{i^*}(2n - \ell^*) \geq 1/3$ and in Section 5.3.2, we reach a contradiction assuming $v_{i^*}(2n - \ell^*) < 1/3$.

Let $\hat{B}_j$ denote the j-th bag at the end of the algorithm. The overall idea is to categorize the bags into different groups and prove an upper bound on the value of each bag ($\hat{B}_j$) for agent i^* depending on which group it belongs to. Since $v_{i^*}(\mathcal{M}) = d$ due to the instance being d-normalized, we get upper and lower bounds on the size of the groups. For example, if we know that for all bags $\hat{B}_j$ in a certain group $v_{i^*}(\hat{B}_j) < 1$, we get the trivial upper bound of $n-1$ on the size of this group since $n = v_{i^*}(\mathcal{M}) = \sum_{j \in [n]} v_{i^*}(\hat{B}_j)$.

Unfortunately, upper bounding the value of the bags is not enough to reach a contradiction in all cases. However, for these cases, we have upper and lower bounds on the size of each group, and in general, we show several additional properties to make it work. For example, we obtain non-trivial upper bounds on the values of certain subsets of goods using the fact that all bundles in a d-MMS partition of agent i^* have value 1 (see Lemmas 5.18 and 5.33).

Note that while we prove certain bounds on the value of the C_k bags for the agents in Section 5.1, these bounds do not carry over to the value of B_k bags. The reason is that we assume the instance is d-normalized, but we have n different B_k bags (in contrast to d different C_k bags). Thus, the difficulty of the analysis is due to translating bounds from C_k's to B_k's.

5.3 1-out-of-4$\lceil n/3 \rceil$ MMS

Algorithm 10 consists of two phases of *initialization* and *bag-filling*. As discussed in Section 5.1, we can assume without loss of generality that $|\mathcal{M}| \geq 2n$. The algorithm first initialize n bags as in (5.3). (See Figure 5.1(b).) Then, in each round j of bag-filling, we keep adding goods of decreasing values to the bag B_j until some agent with no assigned bag values it at least 1. Then, we allocate it to an arbitrary such agent. In the rest of this section, we prove the following theorem, showing the correctness of the algorithm.

Theorem 5.5. *Given any ordered* $4\lceil n/3 \rceil$*-normalized instance, Algorithm 10 returns a* 1*-out-of-*$4\lceil n/3 \rceil$ *MMS allocation.*

To prove Theorem 5.5, it suffices to prove that we never run out of goods in bag-filling. Towards contradiction, assume that the algorithm stops before all agents receive a bundle. Let i^* be an agent with no bundle. Let $\hat{B}_j$ be the j^{th} bundle after bag-filling.

Observation 5.6. *For all* j, k *such that* $j \leq k \leq n$, $v_{i^*}(\hat{B}_j) \leq 1 + v_{i^*}(2n - k + 1)$.

Proof. Let g be the good with the largest index in $\hat{B}_j$. If $g = 2n - j + 1$, $v_{i^*}(\hat{B}_j \setminus \{g\}) = v_{i^*}(j) \leq 1$ by Proposition 5.3((1)). If $g > 2n - j + 1$, meaning that g was added to $\hat{B}_j$ during bag-filling, then $v_{i^*}(\hat{B}_j \setminus \{g\}) < 1$. Otherwise, g would not be added to $\hat{B}_j$. Therefore,

$$\begin{aligned} v_{i^*}(\hat{B}_j) &= v_{i^*}(\hat{B}_j \setminus \{g\}) + v_{i^*}(g) \\ &\leq 1 + v_{i^*}(2n - k + 1). && (v_{i^*}(\hat{B}_j \setminus \{g\}) \leq 1 \text{ and } g \geq 2n - k + 1) \end{aligned}$$

□

Observation 5.7. *For all j, k such that $k \leq j \leq n$, $v_{i^*}(\hat{B}_j) \leq \max(1 + v_{i^*}(2n-k+1), 2v_{i^*}(k))$.*

Proof. First, assume $\hat{B}_j \neq B_j$ and g be the last good added to $\hat{B}_j$. We have $v_{i^*}(\hat{B}_j \setminus \{g\}) < 1$. Otherwise, g would not be added to $\hat{B}_j$. Therefore,

$$\begin{aligned} v_{i^*}(\hat{B}_j) &= v_{i^*}(\hat{B}_j \setminus \{g\}) + v_{i^*}(g) \\ &< 1 + v_{i^*}(2n-k+1). && (v_{i^*}(\hat{B}_j \setminus \{g\}) < 1 \text{ and } g > 2n-k+1) \end{aligned}$$

Now assume $\hat{B}_j = B_j$. We have

$$\begin{aligned} v_{i^*}(\hat{B}_j) &= v_{i^*}(B_j) \\ &= v_{i^*}(j) + v_{i^*}(2n-j+1) \\ &\leq 2v_{i^*}(k). && (2n-j+1 > j \geq k) \end{aligned}$$

Hence, $v_{i^*}(\hat{B}_j) \leq \max(1 + v_{i^*}(2n-k+1), 2v_{i^*}(k))$. □

Observation 5.8. *There exists a bag B_j, such that $v_{i^*}(B_j) < 1$.*

Proof. Otherwise, the algorithm would allocate the remaining bag with the smallest index to agent i^*. □

Let ℓ^* be the smallest such that $v_{i^*}(B_{\ell^*+1}) < 1$. That is, B_{ℓ^*+1} is the leftmost bag in Figure 5.1(b) with a value less than 1 to agent i^*. In Section 5.3.1, we reach a contradiction assuming $v_{i^*}(2n-\ell^*) \geq 1/3$ and prove Theorem 5.9.

Theorem 5.9. *If Algorithm 10 does not allocate a bag to some agent i, then $v_i(2n-\ell^*) < 1/3$ where ℓ^* is smallest such that $v_i(B_{\ell^*+1}) < 1$.*

In Section 5.3.2, we reach a contradiction assuming $v_{i^*}(2n-\ell^*) < 1/3$ and prove Theorem 5.10.

Theorem 5.10. *If Algorithm 10 does not allocate a bag to some agent i, then $v_i(2n-\ell^*) \geq 1/3$ where ℓ^* is the smallest index such that $v_i(B_{\ell^*+1}) < 1$.*

By Theorems 5.10 and 5.9, agent i who receives no bundle by the end of Algorithm 10 does not exist, and Theorem 5.5 follows.

5.3.1 Case: $\mathbf{v_{i^*}(2n - \ell^*) \geq 1/3}$

In this section we assume $v_{i^*}(2n-\ell^*) = 1/3 + x$ for $x \geq 0$. We define $A^+ := \{B_1, B_2, \ldots, B_{\ell^*}\}$; see Figure 5.2.

Observation 5.11. *For all $B_j \in A^+$, $\hat{B}_j = B_j$.*

Proof. For all $B_j \in A^+$, $v_{i^*}(B_j) \geq 1$. Since i^* did not receive any bundle, B_j must have been assigned to some other agent, and no good needed to be added to B_j in bag-filling since there is an agent (namely i^*) with no bag who values B_j at least 1. □

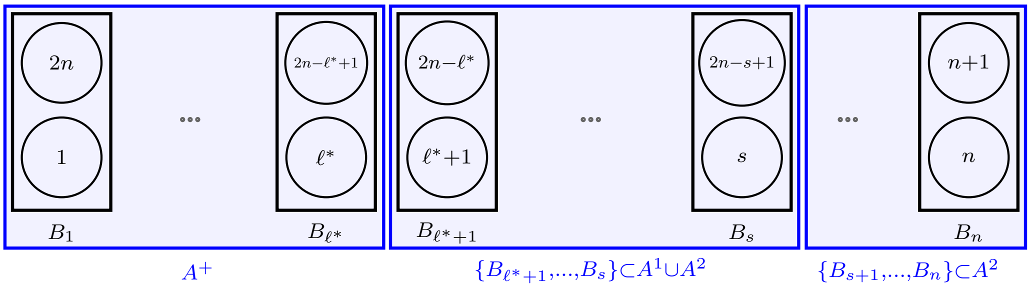


Figure 5.2: An illustration of which group each bag belongs to.

Observation 5.12. *For all $j \geq 2n - \ell^*$, $v_{i^*}(j) < 1/2$.*

Proof. Since $v_{i^*}(B_{\ell^*+1}) = v_{i^*}(\ell^* + 1) + v_{i^*}(2n - \ell^*) < 1$ and $v_{i^*}(2n - \ell^*) \leq v_{i^*}(\ell^* + 1)$, $v_{i^*}(2n - \ell^*) < 1/2$. Also for all $j \geq 2n - \ell^*$, $v_{i^*}(j) \leq v_{i^*}(2n - \ell^*) < 1/2$. □

Corollary 5.13 (of Observation 5.12)**.** *$x < 1/6$.*

Let s be the smallest such that either the algorithm stops at step $s + 1$ or B_{s+1} gets more than one good in bag-filling.

Observation 5.14. *$s \geq \ell^*$.*

Proof. For all $j < \ell^*$, $v_{i^*}(B_{j+1}) \geq 1$. Since i^* did not receive any bundle, B_{j+1}, must have been assigned to another agent. Therefore, the algorithm does not stop at step $j+1$. Also, by Observation 5.11, B_{j+1} gets no good in bag-filling. □

Let A^1 be the set of bags in $\{B_{\ell^*+1}, \dots, B_s\}$ which receive exactly one good in bag-filling. Formally, $A^1 = \{B_j | \ell^* < j \leq s \text{ and } |\hat{B}_j| = 3\}$. Let $A^2 = \{B_1, B_2, \dots, B_n\} \setminus (A^+ \cup A^1)$.

Lemma 5.15. *For all $B_j \in A^2$, $v_{i^*}(\hat{B}_j) < 4/3 - 2x$.*

Proof. We have

$$1 > v_{i^*}(B_{\ell^*+1}) = v_{i^*}(\ell^* + 1) + v_{i^*}(2n - \ell^*) = v_{i^*}(\ell^* + 1) + \frac{1}{3} + x.$$

Hence, $v_{i^*}(\ell^* + 1) < 2/3 - x$. Also, for $B_j \in A^2$, we have

$$\begin{aligned} v_{i^*}(B_j) &= v_{i^*}(j) + v_{i^*}(2n - j + 1) \\ &\leq 2v_{i^*}(\ell^* + 1) && (2n - j + 1 > j \geq \ell^* + 1 \text{ since } B_j \in A^2) \\ &< \frac{4}{3} - 2x. \end{aligned}$$

So if $\hat{B}_j = B_j$, the inequality holds. Now assume $\hat{B}_j \neq B_j$. This implies that $j \geq s + 1$ and the algorithm did not stop at step j before adding a good to B_j. Therefore it did not stop at step $s+1$ before adding a good to B_{s+1} either. Let g be the first good added to B_{s+1}. Since B_{s+1} requires more than one good,

$$\begin{aligned}1 > v_{i^*}(B_{s+1} \cup \{g\}) &= v_{i^*}(s+1) + v_{i^*}(n-s) + v_{i^*}(g) \\ &\geq 2v_{i^*}(2n-\ell^*) + v_{i^*}(g) && (s+1 < n-s \leq 2n-\ell^*) \\ &= \frac{2}{3} + 2x + v_{i^*}(g).\end{aligned}$$

Therefore, $v_{i^*}(g) < 1/3 - 2x$. Now let h be the last good added to bag B_j. We have

$$\begin{aligned}v_{i^*}(\hat{B}_j) &= v_{i^*}(\hat{B}_j \setminus \{h\}) + v_{i^*}(h) \\ &< 1 + v_{i^*}(g) && (v_{i^*}(\hat{B}_j \setminus \{h\}) < 1 \text{ and } v_{i^*}(h) \leq v_{i^*}(g)) \\ &< \frac{4}{3} - 2x.\end{aligned}$$

□

Lemma 5.16. *For all* $B_j \in A^+ \cup A^1$, $v_{i^*}(\hat{B}_j) \leq 4/3 + x$.

Proof. First assume $B_j \in A^+$. We have $j \leq \ell^*$. Also,

$$\begin{aligned}v_{i^*}(\hat{B}_j) &= v_{i^*}(B_j) && (\hat{B}_j = B_j) \\ &= v_{i^*}(j) + v_{i^*}(2n-j+1) \\ &\leq 1 + v_{i^*}(2n-\ell^*) && (v_{i^*}(j) \leq 1 \text{ and } 2n-j+1 > 2n-\ell^*) \\ &= \frac{4}{3} + x.\end{aligned}$$

Now assume $B_j \in A^1$. Let g be the good added to bag B_j in bag-filling. We have,

$$\begin{aligned}v_{i^*}(\hat{B}_j) &= v_{i^*}(B_j) + v_{i^*}(g) \\ &< 1 + v_{i^*}(2n-\ell^*) && (v_{i^*}(B_j) < 1 \text{ and } v_{i^*}(g) \leq v_{i^*}(2n+1) \leq v_{i^*}(2n-\ell^*)) \\ &= \frac{4}{3} + x.\end{aligned}$$

□

Let $|A^1| = 2n/3 + \ell$. Then $|A^2| = n - \ell^* - (2n/3 + \ell) = n/3 - (\ell + \ell^*)$. If $\ell + \ell^* \leq 0$, then $|A^2| \geq n/3$ and hence, by the end of the algorithm, there are at least $n/3$ bags with value less than $4/3 - 2x$ (by Lemma 5.15) and at most $2n/3$ bags with value at most $4/3 + x$ (by Lemma 5.16). Thus,

$$v_{i^*}(\mathcal{M}) < \frac{n}{3}(\frac{4}{3} - 2x) + \frac{2n}{3}(\frac{4}{3} + x) = \frac{4n}{3},$$

which is a contradiction since $v_{i^*}(\mathcal{M}) = 4n/3$. Therefore, $\ell + \ell^* > 0$. Also, note that since $|A^2| = n/3 - (\ell + \ell^*)$, $\ell + \ell^* \leq n/3$.

Observation 5.17. $0 < \ell + \ell^* \leq n/3$.

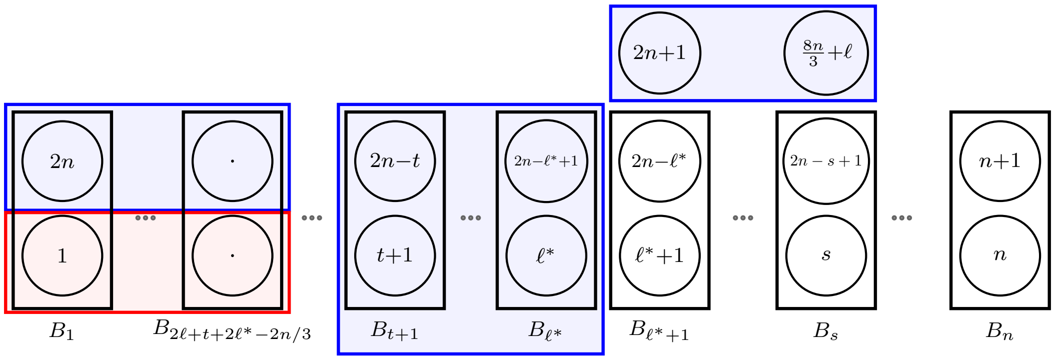


Figure 5.3: The first $2\ell + t + 2\ell^* - 2n/3$ goods are marked with red, and the goods considered in Lemma 5.18 are marked with blue.

Limit the goods in a 1-out-of-$4n/3$ MMS partition $P^{(i^*)} = (P_1^{(i^*)}, \ldots, P_{4n/3}^{(i^*)})$ of agent i^* to $\{1, \ldots, 8n/3 + \ell\}$ and let Q be the set of bags in $P^{(i^*)}$ containing goods $\{1, \ldots, \ell^*\}$. Formally, $Q = \{P_j^{(i^*)} \cap \{1, \ldots, 8n/3 + \ell\} : |P_j^{(i^*)} \cap \{1, \ldots, \ell^*\}| \geq 1\}$. Let t be the number of bags of size 1 in Q.

Lemma 5.18. *Let t be the number of bags of size 1 in $Q = \{P_j^{(i^*)} \cap \{1, \ldots 8n/3 + \ell\} : |P_j^{(i^*)} \cap \{1, \ldots \ell^*\}| \geq 1\}$. Then,*

$$\begin{aligned} v_{i^*}(\{&8n/3 - 2\ell - t - 2\ell^* + 1, \ldots, 8n/3 + \ell\} \\ &\cup \{t+1, \ldots, \ell^*\} \\ &\cup \{2n - \ell^* + 1, \ldots, 2n - t\}) \leq 2\ell^* + \ell - t. \end{aligned}$$

We prove Lemma 5.18 in the end of this section. For now assume that it holds. The goods considered in Lemma 5.18 are marked with blue in Figure 5.3. First, we prove that the goods mentioned in Lemma 5.18 are distinct. To that end, it suffices to prove that $8n/3 - 2\ell - t - 2\ell^* + 1 > 2n - t$. It follows from the fact that $\ell + \ell^* \leq n/3$ (Observation 5.17).

Note that since there are $n/3 - \ell - \ell^*$ bags with value less than $4/3 - 2x$ (namely the bags in A^2), in order to reach a contradiction, it suffices to prove that there exists $3(\ell + \ell^*)$ other bags with total value of at most $4(\ell + \ell^*)$. Since the remaining $2n/3 - 2\ell - 2\ell^*$ bags are of value at most $4/3 + x$ (by Lemma 5.16), we get

$$v_{i^*}(\mathcal{M}) < (\frac{n}{3} - \ell - \ell^*)(\frac{4}{3} - 2x) + (\frac{2n}{3} - 2\ell - 2\ell^*)(\frac{4}{3} + x) + 4(\ell + \ell^*) = \frac{4n}{3} \qquad (5.4)$$

which is a contradiction since $v_{i^*}(\mathcal{M}) = 4n/3$.

Now consider $B = \{\hat{B}_1, \ldots, \hat{B}_{2\ell+t+2\ell^*-2n/3}, \hat{B}_{t+1}, \ldots, \hat{B}_{\ell^*}\} \cup \hat{A}^1$ where $\hat{A}^1$ is the set of bags in A^1 after bag-filling. B consists of $3(\ell + \ell^*)$ bags. Now we prove that $v_{i^*}(\bigcup_{B_j \in B} B_j) \leq 4(\ell + \ell^*)$. We have

$$\begin{aligned} v_{i^*}(\bigcup_{\hat{B}_j \in B} \hat{B}_j) \leq v_{i^*}(\bigcup_{B_j \in A^1} B_j) \\ &+ v_{i^*}(\{1, \ldots, 2\ell + t + 2\ell^* - 2n/3\}) \\ &+ v_{i^*}(\{8n/3 - 2\ell - t - 2\ell^* + 1, \ldots, 8n/3 + \ell\} \\ &\quad \cup \{t + 1, \ldots, \ell^*\} \\ &\quad \cup \{2n - \ell^* + 1, \ldots, 2n - t\}). \end{aligned}$$

We bound the value of the goods marked with different colors in different inequalities.

Observation 5.19. *For all $B_j \in A^1$, $v_{i^*}(B_j) < 1$.*

Since $|A^1| = 2n/3 + \ell$,

$$v_{i^*}(\bigcup_{B_j \in A^1} B_j) < 2n/3 + \ell.$$

Also, since all goods are of value at most 1 to agent i^*,

$$v_{i^*}(\{1, \ldots, 2\ell + t + 2\ell^* - 2n/3\}) \leq 2\ell + t + 2\ell^* - 2n/3.$$

By Lemma 5.18,

$$\begin{aligned} v_{i^*}(&\{8n/3 - 2\ell - t - 2\ell^* + 1, \ldots, 8n/3 + \ell\} \\ &\cup \{t + 1, \ldots, \ell^*\} \\ &\cup \{2n - \ell^* + 1, \ldots, 2n - t\}) \leq 2\ell^* + \ell - t. \end{aligned}$$

By adding all the inequalities, we get

$$v_{i^*}(\bigcup_{\hat{B}_j \in B} \hat{B}_j) \leq 4(\ell + \ell^*).$$

Hence, Inequality (5.4) holds, which is a contradiction. So the case of $v_{i^*}(2n - \ell^*) \geq 1/3$ cannot arise.

Theorem 5.9. *If Algorithm 10 does not allocate a bag to some agent i, then $v_i(2n - \ell^*) < 1/3$ where ℓ^* is smallest such that $v_i(B_{\ell^*+1}) < 1$.*

Proof of Lemma 5.18

To prove Lemma 5.18, we partition the goods considered in this lemma into two parts. These parts are colored red and blue in Figure 5.4. We bound the value of red goods in Lemma 5.22, i.e.,

$$\sum_{2n-\ell^* < j \leq 2n-t} v_{i^*}(j) + \sum_{8n/3-2\ell-t-2\ell^* < j \leq 8n/3-2\ell-2t-\ell^*} v_{i^*}(j) < \ell^* - t,$$

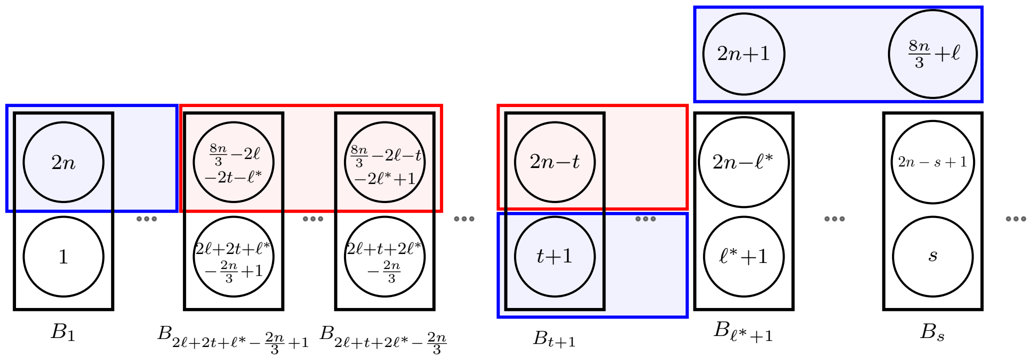


Figure 5.4: The goods considered in in Lemma 5.22 are marked with red and the goods in Lemma 5.23 are marked with blue.

and the value of the blue goods in Lemma 5.23, i.e.,

$$\sum_{t<j\leq \ell^*} v_{i^*}(j) + \sum_{8n/3-2\ell-2t-\ell^*<j\leq 8n/3+\ell} v_{i^*}(j) \leq \ell^* + \ell.$$

Thereafter, we have

$$\begin{aligned}
&v_{i^*}(\{8n/3-2\ell-t-2\ell^*+1,\dots,8n/3+\ell\} \\
&\quad \cup \{t+1,\dots,\ell^*\} \\
&\quad \cup \{2n-\ell^*+1,\dots,2n-t\}) \\
&\quad = \sum_{2n-\ell^*<j\leq 2n-t} v_{i^*}(j) + \sum_{8n/3-2\ell-t-2\ell^*<j\leq 8n/3-2\ell-2t-\ell^*} v_{i^*}(j) \\
&\quad + \sum_{t<j\leq \ell^*} v_{i^*}(j) + \sum_{8n/3-2\ell-2t-\ell^*<j\leq 8n/3+\ell} v_{i^*}(j) \\
&\quad < (\ell^*-t) + (\ell^*+\ell) && \text{(Lemma 5.22 and 5.23)} \\
&\quad = 2\ell^* + \ell - t,
\end{aligned}$$

and Lemma 5.18 follows.

It suffices to prove Lemmas 5.22 and 5.23. In the rest of this section, we prove these two lemmas. Limit the goods in a 1-out-of-$4n/3$ MMS partition of agent i^* to $\{1,\dots,8n/3+\ell\}$ and let R be the set of the resulting bags. Formally, for all $j \in [4n/3]$, $R_j = P_j^{(i^*)} \cap \{1,\dots,8n/3+\ell\}$ and $R = \{R_1,\dots,R_{4n/3}\}$. Without loss of generality, assume $|R_1| \geq |R_2| \geq \dots \geq |R_{4n/3}|$. Recall that t is the number of bags of size 1 in R.

Lemma 5.20. *If there exist t bags of size at most 1 in R, then*

$$\sum_{1\leq j\leq t+\ell} |R_j| \geq 3(t+\ell).$$

Proof. Since R_j's are sorted in decreasing order of their size,

$$\sum_{1\leq j\leq t+\ell} |R_j| \geq (t+\ell)|R_{t+\ell}|.$$

Hence, if $|R_{t+\ell}| \geq 3$, then $\sum_{1\leq j\leq t+\ell} |R_j| \geq 3(t+\ell)$. So assume $|R_{t+\ell}| \leq 2$.

$$\begin{aligned}
\frac{8n}{3} + \ell &= \sum_{1\leq j\leq 4n/3} |R_j| \\
&= \sum_{1\leq j\leq t+\ell} |R_j| + \sum_{t+\ell<j\leq 4n/3-t} |R_j| + \sum_{4n/3-t<j\leq 4n/3} |R_j| \\
&\leq \sum_{1\leq j\leq t+\ell} |R_j| + (\frac{4n}{3} - 2t - \ell)|R_{t+\ell}| + t \\
&\leq \sum_{1\leq j\leq t+\ell} |R_j| + 2(\frac{4n}{3} - 2t - \ell) + t
\end{aligned}$$

Therefore,

$$\sum_{j\in[t+\ell]} |R_j| \geq 3(t+\ell).$$

□

Observation 5.21. $\ell + \ell^* + t \leq 4n/3$.

Proof. By Observation 5.17, $\ell^* + \ell \leq n/3$. Also, $t \leq \ell^* \leq n$. Hence $\ell + \ell^* + t \leq 4n/3$. □

Lemma 5.22.

$$\sum_{2n-\ell^*<j\leq 2n-t} v_{i^*}(j) + \sum_{8n/3-2\ell-t-2\ell^*<j\leq 8n/3-2\ell-2t-\ell^*} v_{i^*}(j) < \ell^* - t.$$

Proof. Let $B' = \{2n-\ell^*+1, \ldots, 2n-t\} \cup \{8n/3-2\ell-t-2\ell^*+1, \ldots, 8n/3-2\ell-2t-\ell^*\}$. $|B'| = 2(\ell^* - t)$ and by Observation 5.12 for all goods $g \in B'$, $v_{i^*}(g) < 1/2$. Therefore, $v_{i^*}(B') < \ell^* - t$. □

Lemma 5.23.

$$\sum_{t<j\leq\ell^*} v_{i^*}(j) + \sum_{8n/3-2\ell-2t-\ell^*<j\leq 8n/3+\ell} v_{i^*}(j) \leq \ell^* + \ell.$$

Proof. Recall that $\{R_1, \ldots, R_{4n/3}\}$ is the set of bags in the 1-out-of-$4n/3$ MMS partition of agent i^* after removing goods $\{8n/3+\ell+1, \ldots, m\}$. Moreover, we know exactly t of these bags have size 1. If there is a bag $R_j = \{g\}$ for $g > t$, there must be a good $g' \in [t]$ such that $g' \in R_{j'}$ and $|R_{j'}| > 1$. Swap the goods g and g' between R_j and $R_{j'}$ as long as such good g exists. Note that $v_{i^*}(R_{j'})$ can only decrease and $v_{i^*}(R_j) = v_{i^*}(g') \leq 1$. Therefore, in the end of this process for all $j \in [4n/3]$, $v_{i^*}(R_j) \leq 1$ and we can assume bags containing goods $1, \ldots, t$ are of size 1 and bags containing goods $t+1, \ldots, \ell^*$ are of a size more than 1. Recall that $|R_1| \geq \ldots \geq |R_{4n/3}|$. Let T_j be the bag that contains good j.

Consider the bags $B = \{R_1, \ldots, R_{t+\ell}\} \cup \{T_{t+1}, \ldots, T_{\ell^*}\}$. If $|B| < \ell^* + \ell$, keep adding a bag with the largest number of goods to B until there are exactly $\ell^* + \ell$ bags in B. First we show that B contains at least $3\ell + 2\ell^* + t$ goods. Namely,

$$\sum_{S \in B} |S| \geq 3\ell + 2\ell^* + t.$$

By Lemma 5.20, $\sum_{1 \leq j \leq t+\ell} |R_j| \geq 3(t+\ell)$. If all the remaining $\ell^* - t$ bags in $B \setminus \{R_1, \ldots, R_{t+\ell}\}$ are of size 2, then $\sum_{S \in B} |S| \geq 3(t+\ell) + 2(\ell^* - t) = 3\ell + 2\ell^* + t$. Otherwise, there is a bag in B of size at most 1; hence, all bags outside B are also of size at most 1. So we have

$$\begin{aligned} \frac{8n}{3} + \ell &= \sum_{S \in B} |S| + \sum_{S \notin B} |S| \\ &\leq \sum_{S \in B} |S| + (\frac{4n}{3} - \ell^* - \ell). \end{aligned}$$

Therefore,

$$\begin{aligned} \sum_{S \in B} |S| &\geq 4n/3 + 2\ell + \ell^* \\ &\geq 3\ell + 2\ell^* + t. && \text{(Observation 5.21)} \end{aligned}$$

Note that the goods $\{t+1, \ldots, \ell^*\}$ are contained in B and moreover, B contains at least $3\ell + 2\ell^* + t - (\ell^* - t) = 3\ell + 2t + \ell^*$ other goods. Therefore,

$$\begin{aligned} \ell^* + \ell &\geq \sum_{S \in B} v_{i^*}(S) \\ &\geq \sum_{t < j \leq \ell^*} v_{i^*}(j) + \sum_{8n/3 - 2\ell - 2t - \ell^* < j \leq 8n/3 + \ell} v_{i^*}(j). \end{aligned}$$

The last inequality follows because we used the $3\ell + 2t + \ell^*$ lowest valued goods in $[8n/3 + \ell]$. □

5.3.2 Case: $\mathbf{v_{i^*}(2n - \ell^*) < 1/3}$

Let r^* be largest such that $v_{i^*}(B_{r^*}) < 1$. That is, B_{r*} is the rightmost bag in Figure 5.1(b) with a value less than 1 to agent i^*.

Lemma 5.24. *If $v_{i^*}(2n - r^* + 1) \leq 1/3$, then $r^* < 2n/3$.*

Proof. Let $x = 1/3 - v_{i^*}(2n - \ell^*)$. Since $1 > v_{i^*}(B_{r^*}) = v_{i^*}(r^*) + v_{i^*}(2n - r^* + 1)$, we have $v_{i^*}(r^*) < 2/3 + x$. By Observation 5.6, for all $j \leq r^*$, $v_{i^*}(\hat{B}_j) \leq 4/3 - x$. Also, by Observation 5.7, for all $j > r^*$, $v_{i^*}(\hat{B}_j) < 4/3 + 2x$. Hence, we have

$$\begin{aligned} \frac{4n}{3} &= v_{i^*}(\mathcal{M}) \\ &= \sum_{j \leq r^*} v_{i^*}(\hat{B}_j) + \sum_{j > r^*} v_{i^*}(\hat{B}_j) \\ &< r^*(\frac{4}{3} - x) + (n - r^*)(\frac{4}{3} + 2x) \\ &= \frac{4n}{3} + x(2n - 3r^*). \end{aligned}$$

Therefore, $r^* < 2n/3$. □

Lemma 5.25. $v_{i^*}(2n - r^* + 1) > 1/3$.

Proof. Towards contradiction, assume $v_{i^*}(2n - r^* + 1) = 1/3 - x$ for $x \geq 0$. By Lemma 5.24, $r^* < 2n/3$.

Claim 5.26. $\sum_{j>r^*} v_{i^*}(\hat{B}_j) < \frac{10n}{9} - r^* + \frac{2nx}{3}$.

Proof. Note that by the definition of r^*, for all $j > r^*$, $\hat{B}_j = B_j$. By Lemma 5.4, $v_{i^*}(\{2n/3 + r^* + 1, \dots, 2n - r^*\}) \leq 2n/3 - r^*$. Also since $v_{i^*}(r^*) < 2/3 + x$, $v_{i^*}(\{r^* + 1, \dots, 2n/3 + r^*\}) \leq \frac{2n}{3}(\frac{2}{3} + x)$. In total, we get

$$\begin{aligned}
\sum_{j>r^*} v_{i^*}(\hat{B}_j) &= \sum_{j>r^*} v_{i^*}(B_j) \\
&= v_{i^*}(\{r^* + 1, \dots, 2n/3 + r^*\}) + v_{i^*}(\{2n/3 + r^* + 1, \dots, 2n - r^*\}) \\
&< \frac{2n}{3}(\frac{2}{3} + x) + \frac{2n}{3} - r^* \\
&= \frac{10n}{9} - r^* + \frac{2nx}{3}.
\end{aligned}$$

Therefore, Claim 5.26 holds. ■

We have

$$\begin{aligned}
\frac{4n}{3} &= v_{i^*}(\mathcal{M}) \\
&= \sum_{j \leq r^*} v_{i^*}(\hat{B}_j) + \sum_{j>r^*} v_{i^*}(\hat{B}_j) \\
&< r^*(\frac{4}{3} - x) + \frac{10n}{9} - r^* + \frac{2nx}{3} && \text{(Observation 5.6 and Claim 5.26)} \\
&= r^*(\frac{1}{3} - x) + \frac{10n}{9} + \frac{2nx}{3}.
\end{aligned}$$

Thus,

$$\begin{aligned}
\frac{2n}{9} &< r^*(\frac{1}{3} - x) + \frac{2n}{3}(x) \\
&\leq \frac{2n}{3} \cdot \frac{1}{3}, && (r^* \leq 2n/3 \text{ by Lemma 5.24})
\end{aligned}$$

which is a contradiction. Hence, $v_{i^*}(2n - r^* + 1) > 1/3$. □

Recall that ℓ^* is the smallest such that $v_{i^*}(B_{\ell^*+1}) < 1$, i.e., B_{ℓ^*+1} is the leftmost bag in Figure 5.1(b) with value less than 1 to agent i^*. Let ℓ be largest such that $v_{i^*}(B_\ell) < 1$ and $v_{i^*}(2n - \ell + 1) \leq 1/3$. Since $v_{i^*}(B_{\ell^*+1}) < 1$ and $v_{i^*}(2n - \ell^*) < 1/3$, such an index exists and $\ell \geq \ell^* + 1$. Also, let r be smallest such that $v_{i^*}(B_{r+1}) < 1$ and $v_{i^*}(2n - r) \geq 1/3$. Again, since $v_{i^*}(B_{r^*}) < 1$ and $v_{i^*}(2n - r^* + 1) > 1/3$, such an index exists. We set $x := 1/3 - v_{i^*}(2n - \ell + 1)$ and $y := v_{i^*}(2n - r) - 1/3$. See Figure 5.5.

Observation 5.27. *For all* $\ell < j \leq r$, $1 \leq v_{i^*}(\hat{B}_j) < 1 + x + y$.

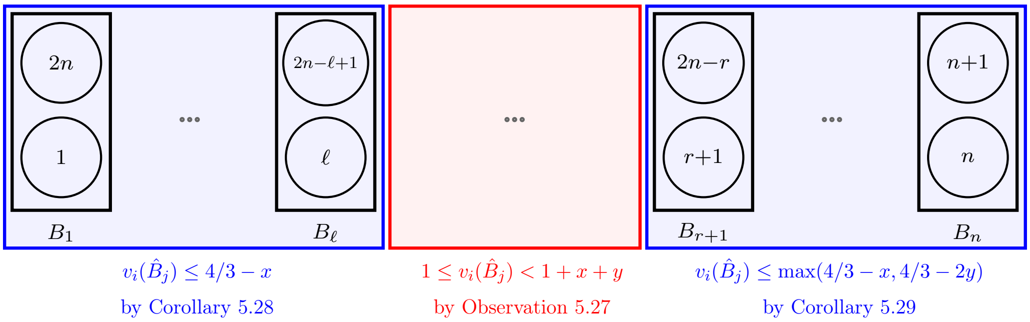


Figure 5.5: $v_{i^*}(2n-\ell+1) = 1/3 - x$ and $v_{i^*}(2n-r) = 1/3 + y$.

Proof. Note that by definition of ℓ and r, for all $\ell < j \le r$, $v_{i^*}(B_j) \ge 1$. Therefore, $\hat{B}_j = B_j$. Also,

$$\begin{aligned} v_{i^*}(B_j) &= v_{i^*}(j) + v_{i^*}(2n-j+1) \\ &\le v_{i^*}(\ell) + v_{i^*}(2n-r) && (\ell < j \text{ and } 2n-r < 2n-j+1) \\ &< (\frac{2}{3}+x) + (\frac{1}{3}+y) && (v_{i^*}(B_\ell) < 1 \text{ and } v_{i^*}(B_{r+1}) < 1) \\ &= 1 + x + y. \end{aligned}$$

□

Corollary 5.28 (of Observation 5.6)**.** *For all $j \le \ell$, $v_{i^*}(\hat{B}_j) \le 4/3 - x$.*

Corollary 5.29 (of Observation 5.7)**.** *For all $j > r$, $v_{i^*}(\hat{B}_j) \le \max(4/3 - x, 4/3 - 2y)$.*

Observation 5.30. $x < 1/3$.

Proof. Towards a contradiction, assume $x = 1/3$. Therefore, $v_{i^*}(2n-\ell+1) = 0$. Let $k < 2n-\ell+1$ be the number of goods with a value larger than 0 to agent i^*. Consider $(P_1^i \cap [k], \ldots, P_{4n/3}^{(i^*)} \cap [k])$. There are at least ℓ many indices j such that, $|P_j^{(i^*)} \cap [k]| = 1$. Since $\mathcal{I}$ is $4n/3$-normalized, $v_{i^*}(1) = \ldots = v_{i^*}(\ell) = 1$ which is a contradiction with $v_{i^*}(B_{\ell^*+1}) < 1$ since $\ell^* + 1 \le \ell$. □

Observation 5.31. $y < 1/6$.

Proof. We have $1/3 + y = v_{i^*}(2n-r) \le v_{i^*}(B_r)/2 < 1/2$. Thus, $y < 1/6$. □

Lemma 5.32. $r - \ell > 2n/3$.

Proof. If $x + y \le 1/3$, then by Corollaries 5.28 and 5.29 and Observation 5.27, for all $t \in [n]$ we have $v_{i^*}(\hat{B}_t) \le 4/3$ and for at least one bag this value is less than 1 by

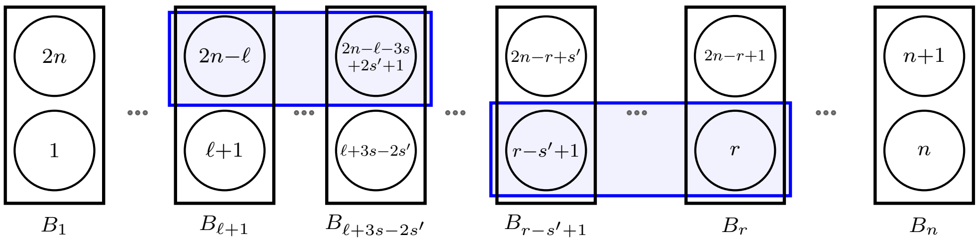


Figure 5.6: The goods considered in Lemma 5.33 are marked with blue.

Observation 5.8. Therefore, $v_{i^*}(\mathcal{M}) < 4n/3$, which is a contradiction. Thus, $x + y > 1/3$. We have

$$\begin{aligned}
\frac{4n}{3} &= v_{i^*}(\mathcal{M}) \\
&= \sum_{j\le\ell} v_{i^*}(\hat{B}_j) + \sum_{\ell<j\le r} v_{i^*}(\hat{B}_j) + \sum_{j>r} v_{i^*}(\hat{B}_j) \\
&\le \ell(\frac{4}{3} - x) + (r-\ell)(1+x+y) + (n-r)\max(\frac{4}{3} - x, \frac{4}{3} - 2y) \\
&\qquad\qquad \text{(Corollaries 5.28 and 5.29 and Observation 5.27)} \\
&\le (r-\ell)(1+x+y) + (n-r+\ell)\max(\frac{4}{3} - x, \frac{4}{3} - 2y) \\
&= \frac{4n}{3} + (r-\ell)(x+y-\frac{1}{3}) - (n-r+\ell)\min(x, 2y).
\end{aligned}$$

Therefore, $(r-\ell)(x+y-1/3) \ge (n-r+\ell)\min(x,2y)$. By Observation 5.30, $x < 1/3$ and thus, we have $x + y - 1/3 < y$. Also, since $y < 1/6$ (by Observation 5.31), we have $x+y-1/3 < x-1/6 < x/2$. Thus, $x+y-1/3 < \min(x,2y)/2$. Hence, $r-\ell > 2(n-r+\ell)$ and therefore, $r - \ell > 2n/3$. □

Let $r - \ell = 2n/3 + s$. Recall that $P^{(i^*)} = (P^{(i^*)}_1, \dots, P^{(i^*)}_{4n/3})$ is an $(4n/3)$-MMS partition of $\mathcal{M}$ for agent i^*. Since i^* is fixed, we use $P = (P_1, \dots, P_{4n/3})$ instead for ease of notation. For all $j \in [4n/3]$, let g_j be the good with the smallest index (and hence the largest value) in P_j. Without loss of generality, assume $g_1 < g_2 < \ldots < g_{4n/3}$. Observe that $\{1, \dots, r\} \subseteq \cup_{k\in[r]} P_k$. Let S' be the set of goods in $\{r+1, \dots, 2n-\ell\}$ that appear in the first r bags in P. Formally, $S' = \{g \in \{r+1, \dots, 2n-\ell\} \mid g \in \cup_{j\in[r]} P_j\}$. Let $s' := \min(|S'|, s)$.

Lemma 5.33. $v_{i^*}(\{r-s'+1, \dots, r\} \cup \{2n-\ell-3s+2s'+1, \dots, 2n-\ell\}) \le s$.

We prove Lemma 5.33 in the end of this section. For now assume that it holds.

The goods considered in Lemma 5.33 are marked with blue in Figure 5.6. Since $v_{i^*}(\ell) < 1 - v_{i^*}(2n-\ell+1) = 2/3 + x$, we have

$$v_{i^*}(\{\ell+1, \dots, r-s'\}) < (\frac{2n}{3} + s - s')(\frac{2}{3} + x). \tag{5.5}$$

Also, since $v_{i^*}(2n-r+1) = 1/3 + y$,

$$v_{i^*}(\{2n-r+1, \dots, 2n-\ell-3s+2s'\}) \le (\frac{2n}{3} - 2s + 2s')(\frac{1}{3} + y). \tag{5.6}$$

Therefore,

$$\begin{aligned}
\sum_{\ell<j\le r} v_{i^*}(\hat{B}_j) = \sum_{\ell<j\le r} v_{i^*}(B_j) &= v_{i^*}(\{\ell+1,\dots,r\}\cup\{2n-r+1,\dots,2n-\ell\}) \\
&= v_{i^*}(\{\ell+1,\dots,r-s'\}) \\
&\quad + v_{i^*}(\{r-s'+1,\dots,r\}\cup\{2n-\ell-3s+2s'+1,\dots,2n-\ell\}) \\
&\quad + v_{i^*}(\{2n-r+1,\dots,2n-\ell-3s+2s'\}) \\
&< (\frac{2n}{3}+s-s')(\frac{2}{3}+x)+s+(\frac{2n}{3}-2s+2s')(\frac{1}{3}+y) \\
&\qquad \text{(Inequalities (5.5) and (5.6) and Lemma 5.33)} \\
&= \frac{2n}{3}(1+x+y)+(s-s')(x-2y)+s.
\end{aligned}$$

Thus,

$$\begin{aligned}
\frac{4n}{3} = v_{i^*}(\mathcal{M}) &= \sum_{j\le \ell} v_{i^*}(\hat{B}_j) + \sum_{\ell<j\le r} v_{i^*}(\hat{B}_j) + \sum_{j>r} v_{i^*}(\hat{B}_j) \\
&< (\ell+n-r)\max(\frac{4}{3}-x,\frac{4}{3}-2y)+\frac{2n}{3}(1+x+y)+(s-s')(x-2y)+s \\
&\qquad \text{(Corollaries 5.28 and 5.29)} \\
&= (\frac{n}{3}-s)\max(\frac{4}{3}-x,\frac{4}{3}-2y)+\frac{2n}{3}(1+x+y)+(s-s')(x-2y)+s.
\end{aligned}$$

If $x \le 2y$, then by replacing $\max(4/3-x, 4/3-2y)$ with $4/3-x$ in the above inequality, we get

$$\begin{aligned}
\frac{4n}{3} &< (\frac{n}{3}-s)(\frac{4}{3}-x)+\frac{2n}{3}(1+x+y)+(s-s')(x-2y)+s \\
&\le \frac{n}{3}(\frac{4}{3}-x)+\frac{2n}{3}(1+x+y)+(s-s')(x-2y) && (4/3-x\ge 1) \\
&\le \frac{n}{3}(\frac{10}{3}+x+2y) && ((s-s')(x-2y)\le 0) \\
&< \frac{4n}{3}, && (x\le 1/3 \text{ and } y<1/6)
\end{aligned}$$

which is a contradiction. If $2y < x$, by replacing $\max(4/3-x, 4/3-2y)$ with $4/3-2y$, we get

$$\begin{aligned}
\frac{4n}{3} &< (\frac{n}{3}-s)(\frac{4}{3}-2y)+\frac{2n}{3}(1+x+y)+(s-s')(x-2y)+s \\
&= \frac{n}{3}(\frac{10}{3}+2x)-s(\frac{1}{3}-x)-s'(x-2y) \\
&\le \frac{n}{3}(\frac{10}{3}+2x) && (x\le 1/3 \text{ and } x>2y) \\
&\le \frac{4n}{3}, && (x\le 1/3)
\end{aligned}$$

which is again a contradiction. Therefore, it is not possible that $v_{i^*}(2n-\ell^*) < 1/3$. Thus, Theorem 5.10 follows.

Theorem 5.10. *If Algorithm 10 does not allocate a bag to some agent i, then $v_i(2n-\ell^*) \ge 1/3$ where ℓ^* is the smallest index such that $v_i(B_{\ell^*+1}) < 1$.*

Proof of Lemma 5.33

The main idea of the proof is as follows. Recall that $s' = \min(|S'|, s)$. We consider two cases for s'. If $s' = s$, then in order to prove Lemma 5.33, we must prove

$$v_{i^*}(\{r - s' + 1, \dots, r\} \cup \{2n - \ell - s' + 1, \dots, 2n - \ell\}) \leq s',$$

which is what we do in Claim 5.35. In case $s' = |S'|$, we prove

$$v_{i^*}(\{r - s' + 1, \dots, r\}) + v_{i^*}(S') \leq s'$$

in Claim 5.36 and

$$v_{i^*}(\{2n - \ell - 3s + 2s' + 1, \dots, 2n - \ell\}) - v_{i^*}(S') \leq s - s'$$

in Claim 5.37. Adding the two sides of the inequalities implies Lemma 5.33.

Note that $\{1, \dots, r\} \cup S' \subseteq P_1 \cup \ldots \cup P_r$. For $j \in [r]$, let $Q_j = P_j \cap (\{1, \dots, r\} \cup S')$. We begin with proving the following claim.

Claim 5.34. *There are s' many sets like $Q_{j_1}, \dots, Q_{j_{s'}}$ such that $|\cup_{k\in[s']} Q_{j_k}| \geq 2s'$ and $|\cup_{k\in[s']} Q_{j_k} \cap \{1, \dots, r\}| \geq s'$.*

Proof. If $s' = 0$, the claim trivially holds. Thus, assume $s' \geq 1$. By induction, we prove that for any $t \leq s'$, there are t many sets like $Q_{j_1}, \dots, Q_{j_t}$ such that $|\cup_{k\in[t]} Q_{j_k}| \geq 2t$ and $|\cup_{k\in[t]} Q_{j_k} \cap \{1, \dots, r\}| \geq t$.

Induction basis: $t = 1$. If there exists Q_k such that $|Q_k \cap \{1, \dots, r\}| \geq 2$, let $j_1 = k$. Otherwise, for all $k \in [r]$, we have $|Q_k \cap \{1, \dots, r\}| = 1$. Since $s' \geq 1$, there must be an index k such that $|Q_k \cap S'| \geq 1$. Let $j_1 = k$.

Induction assumption: There are t many sets like $Q_{j_1}, \dots, Q_{j_t}$ such that $|\cup_{k\in[t]} Q_{j_k}| \geq 2t$ and $|\cup_{k\in[t]} Q_{j_k} \cap \{1, \dots, r\}| \geq t$.

Now for $t + 1 \leq s'$, we prove that there are $t + 1$ many sets like $Q_{j_1}, \dots, Q_{j_{t+1}}$ such that $|\cup_{k\in[t+1]} Q_{j_k}| \geq 2t + 2$ and $|\cup_{k\in[t+1]} Q_{j_k} \cap \{1, \dots, r\}| \geq t + 1$.

Case 1: $|\cup_{k\in[t]} Q_{j_k}| \geq 2t + 2$: If $|\cup_{k\in[t]} Q_{j_k} \cap \{1, \dots, r\}| \geq t + 1$, set $j_{t+1} = k$ for an arbitrary $k \in [r] \setminus \{j_1, \dots, j_t\}$. Otherwise, set $j_{t+1} = k$ for an index $k \in [r] \setminus \{j_1, \dots, j_t\}$ such that $|Q_k \cap \{1, \dots, r\}| \geq 1$.

Case 2: $|\cup_{k\in[t]} Q_{j_k}| = 2t + 1$: If there exists $k \in [r] \setminus \{j_1, \dots, j_t\}$, such that $|Q_k \cap [r]| \geq 1$, set $j_{t+1} = k$. Otherwise, set $j_{t+1} = k$ for any $k \in [r] \setminus \{j_1, \dots, j_t\}$ such that $|Q_k| \geq 1$. Since $|\cup_{j\in[r]} Q_j| \geq r + s' > 2t + 1$, such k exists.

Case 3. $|\cup_{k\in[t]} Q_{j_k}| = 2t$ and $|\cup_{k\in[t]} Q_{j_k} \cap \{1, \dots, r\}| \geq t + 1$: $|\cup_{k\in[r]\setminus\{j_1,\dots,j_t\}} Q_{j_k}| \geq r + s' - 2t > r - t$. Therefore, by pigeonhole principle, there exists an index $k \in [r] \setminus \{j_1, \dots, j_t\}$ such that $|Q_k| \geq 2$. Set $j_{t+1} = k$.

Case 4. $|\cup_{k\in[t]} Q_{j_k}| = 2t$ **and** $|\cup_{k\in[t]} Q_{j_k} \cap \{1,\dots,r\}| = t$**:** If there exists $k \in [r]\setminus\{j_1,\dots,j_t\}$, such that $|Q_k \cap [r]| \geq 2$, set $j_{t+1} = k$. Otherwise, for all $k \in [r]\setminus\{j_1,\dots,j_t\}$, $|Q_k \cap [r]| = 1$ since $|\cup_{k\in[t]} Q_{j_k} \cap \{1,\dots,r\}| = t$ and $|\cup_{k\in[r]} Q_{j_k} \cap \{1,\dots,r\}| = r$. Set $j_{t+1} = k$ for any $k \in [r]\setminus\{j_1,\dots,j_t\}$, such that $|Q_k \cap S'| \geq 1$. Since $|\cup_{j\in[r]} Q_j \cap S'| \geq s' > t$, such k exists. □

Now we prove Claim 5.35.

Claim 5.35. $v_{i^*}(\{r-s'+1,\dots,r\} \cup \{2n-\ell-s'+1,\dots,2n-\ell\}) \leq s'$.

Proof. Let Q^1 be the set of s' most valuable goods in $\cup_{k\in[s']} Q_{j_k}$ and let Q^2 be the set of s' least valuable goods in $\cup_{k\in[s']} Q_{j_k}$. Since $|\cup_{k\in[s']} Q_{j_k}| \geq 2s'$, $Q^1 \cap Q^2 = \emptyset$. Also, $|\cup_{k\in[s']} Q_{j_k} \cap \{1,\dots,r\}| \geq s'$. Thus, $v_{i^*}(Q^1) \geq v_{i^*}(\{r-s'+1,\dots,r\})$. Moreover, $v_{i^*}(Q^2) \geq v_{i^*}(\{2n-\ell-s'+1,\dots,2n-\ell\})$. Hence,

$$\begin{aligned} s' &= \sum_{k\in[s']} v_{i^*}(P_{j_k}) \\ &\geq \sum_{k\in[s']} v_{i^*}(Q_{j_k}) \\ &\geq v_{i^*}(\{r-s'+1,\dots,r\} \cup \{2n-\ell-s'+1,\dots,2n-\ell\}). \end{aligned}$$

□

Note that in case $s' = s$, Claim 5.35 implies Lemma 5.33. Therefore, from now on, we assume $s' = |S'| < s$.

Claim 5.36. $v_{i^*}(\{r-s'+1,\dots,r\}) + v_{i^*}(S') \leq s'$.

Proof. The proof is similar to the proof of Claim 5.35. Let Q^1 be the set of s' most valuable goods in $\cup_{k\in[s']} Q_{j_k}$ and let Q^2 be the set of s' least valuable goods in $\cup_{k\in[s']} Q_{j_k}$. Since $|\cup_{k\in[s']} Q_{j_k}| \geq 2s'$, $Q^1 \cap Q^2 = \emptyset$. Also, $|\cup_{k\in[s']} Q_{j_k} \cap \{1,\dots,r\}| \geq s'$. Thus, $v_{i^*}(Q^1) \geq v_{i^*}(\{r-s'+1,\dots,r\})$. Moreover, $v_{i^*}(Q^2) \geq v_{i^*}(S')$ since $s' = |S'|$. Hence,

$$\begin{aligned} s' &= \sum_{k\in[s']} v_{i^*}(P_{j_k}) \\ &\geq \sum_{k\in[s']} v_{i^*}(Q_{j_k}) \\ &\geq v_{i^*}(\{r-s'+1,\dots,r\} \cup S') \\ &= v_{i^*}(\{r-s'+1,\dots,r\}) + v_{i^*}(S'). \end{aligned}$$

□

Claim 5.37. $v_{i^*}(\{2n-\ell-3s+2s'+1,\dots,2n-\ell\}) - v_{i^*}(S') \leq s-s'$.

Proof. Note that by definition of S', the $2n-\ell-r-s' = 8n/3-2r+s-s'$ goods in $\{r+1,\dots,2n-\ell\} \setminus S'$ are in $P_{r+1} \cup \ldots \cup P_{4n/3}$. Now for $j \in [4n/3-r]$, let $R_j = P_{j+r} \cap \{r+1,\dots,2n-\ell\} \setminus S'$. Assume $|R_{j_1}| \geq \ldots \geq |R_{j_{4n/3-r}}|$. We prove

$$\sum_{k\leq s-s'} |R_{j_k}| \geq 3(s-s'). \tag{5.7}$$

If $|R_{j_{s-s'+1}}| \geq 3$, Inequality (5.7) holds. Otherwise, we have

$$\begin{aligned}
\frac{8n}{3} - 2r + s - s' &= \sum_{k\in[4n/3-r]} |R_{j_k}| \\
&= \sum_{k\leq s-s'} |R_{j_k}| + \sum_{s-s'<k\leq 4n/3-r} |R_{j_k}| \\
&\leq \sum_{k\leq s-s'} |R_{j_k}| + 2(\frac{4n}{3} - r - s + s'). \qquad (|R_{j_k}| \leq 2 \text{ for } k > s-s')
\end{aligned}$$

Thus, $\sum_{k\in[s-s']} |R_{j_k}| \geq 3(s-s')$. We have

$$\begin{aligned}
s - s' &= \sum_{k\in[s-s']} v_{i^*}(P_{j_k+r}) \\
&\geq \sum_{k\in[s-s']} v_{i^*}(R_{j_k}) \\
&\geq v_{i^*}(\{2n-\ell-3s+2s'+1,\dots,2n-\ell\}) - v_{i^*}(S'). \\
&\qquad (|\cup_{k\in[s-s']} R_{j_k}| \geq 3(s-s') \text{ and } |S'| = s')
\end{aligned}$$

□

Claims 5.36 and 5.37 imply Lemma 5.33.

Recap of Section 5.3: To show that a 1-out-of-$4n/3$ MMS allocation exists, it suffices to prove that we never run out of goods for bag-filling in Algorithm 10. Towards contradiction, we assumed that the algorithm stops before agent i^* receives a bundle. By Observation 5.8, a bag with a value less than 1 for agent i^* exists. Let ℓ^* be the smallest such that $v_{i^*}(B_{\ell^*+1}) < 1$. In Section 5.3.2, we reached a contradiction assuming $v_{i^*}(2n-\ell^*) < 1/3$ and proved Theorem 5.10. In Section 5.3.1, we reached a contradiction assuming $v_{i^*}(2n-\ell^*) \geq 1/3$ and proved Theorem 5.9. Therefore, no such agent i^* exists, and all agents receive a bag by the end of Algorithm 10. Theorem 5.5 follows.

PART II

Envy-Based Fairness Notions

CHAPTER 6

EFX for Three Agents

Plaut and Roughgarden [59] first showed the existence of EFX allocations when there are two agents, using the *cut and choose protocol*. The existence of EFX allocations gets notoriously difficult with three or more agents. The existence of EFX allocations for three agents with *additive valuations* was shown by Chaudhury et al. [27]. After that, Berger et al. [15] showed the existence of EFX allocations with three agents when agents have *cancelable valuation functions* – a class that subsumes additive, budget-additive, unit demand, and multiplicative valuation functions. However, this technique does not extend, as soon as one of the agents has a general monotone valuation function. Despite its fundamentality and ongoing efforts, the existence of EFX allocations with three agents under general valuation functions remains elusive. Plaut and Roughgarden [59] remarked: "We suspect that at least for general valuations, there exist instances where no EFX allocation exists, and it may be easier to find a counterexample in that setting". In this chapter, we make progress on this problem. We show the existence of EFX allocations, when *two* agents have general monotone valuation functions.

Theorem 6.1. *EFX allocations exist with three agents as long as at least one agent has an additive valuation function (the other two agents have general monotone valuation functions).*

In fact, our proof gives a stronger version of Theorem 6.1: we can show the existence of EFX allocations when two agents have general monotone valuation functions and one of the agents has an *MMS-feasible valuation function* – a valuation class that strictly generalizes cancelable valuation functions – definitions and properties are described in Section 2.2. Thus, we strictly generalize the result in [15].

Theorem 6.2. *EFX allocations exist with three agents as long as at least one agent has an MMS-feasible valuation function.*

We briefly remark on our technique to prove Theorem 6.2 and how it crucially differs from the existing techniques in [15, 27, 56]. The algorithms in [15, 27, 56] move in the space of partial EFX allocations (where not all goods are allocated) iteratively improving the vector $\langle v_1(X_1), v_2(X_2), v_3(X_3)\rangle$ lexicographically, where $v_i(\cdot)$ is the valuation function of agent i. However, [28] exhibits an instance with four agents and a partial EFX allocation X, such that in any complete EFX allocation X', $v_1(X'_1) < v_1(X_1)$, i.e., agent 1 (which is the highest priority agent) is better off in X than in any complete EFX allocation. This implies that if the algorithm starts with allocation X or reaches X at some point, it cannot output a complete EFX allocation. This necessitates the study of a different approach for the existence of EFX allocations. Our algorithm moves in the space of complete allocations (instead of partial allocations), iteratively improving a certain potential as long as the current allocation is not EFX. Furthermore, this proof turns out to be simpler and significantly shorter than the ones in [15, 27], as it does not use the notions of champions, champion-graphs, half-bundles, and even the envy-graph.

6.1 Notations and Tools

Recall the definition of MMS-feasible valuations.

Definition 6.3. *A valuation function $v : 2^{\mathcal{M}} \to \mathbb{R}_{\geq 0}$ is MMS-feasible if for every subset of goods $S \subseteq \mathcal{M}$ and every partitions $A = (A_1, A_2)$ and $B = (B_1, B_2)$ of S, we have*

$$\max(v(B_1), v(B_2)) \geq \min(v(A_1), v(A_2)).$$

By Lemma 2.32, we know the class of MMS-feasible valuation functions is a strict superclass of cancelable valuation functions.

Without loss of generality (by Lemma 2.27), in this chapter we assume the instance is non-degenerate.

6.2 EFX Existence beyond Additivity

In this section, we present an algorithmic proof for the existence of EFX allocations when there are three agents with valuations more general than additive. The main takeaway of our algorithm is that it does not require the sophisticated concepts introduced by the techniques in [27, 29] that rely on improving a potential function while moving in the space of partial EFX allocations. In fact, our algorithm does not even require the concept of envy-graph, which is a very fundamental concept used by the algorithms in [27, 29] and also in [55, 59] to prove the existence of weaker relaxations of envy-freeness (like 1/2-EFX and EF1).

First, we sketch our algorithm. The crucial idea in our technique is to move in the space of partitions (of the goods set), improving a certain potential as long as we cannot find an EFX allocation from the current partition, i.e., we cannot find a *complete* allocation of the bundles in the partition such that the EFX property is satisfied. In particular, we always maintain a partition $X = (X_1, X_2, X_3)$ such that (i) agent 1 finds X_1 and X_2 EFX-feasible and (ii) at least one of agent 2 and agent 3 finds X_3 EFX-feasible. Note that such allocations always exist: Agent 1 can determine a partition such that all bundles are EFX-feasible for her (such a partition is possible as agent 1 can run the algorithm in [59] to find an EFX allocation assuming all three agents have agent 1's valuation function, thereby making all bundles EFX-feasible for her). We call agent 2's favorite bundle in the partition X_3 (which is obviously EFX-feasible for her) and the remaining bundles X_1 and X_2 arbitrarily. Then, we have a partition that satisfies the invariant.

Note that if any one of agent 2 or 3 finds one of X_1 or X_2 EFX-feasible, then we easily get an EFX allocation. Indeed, assume without loss of generality that agent 2 finds X_3 EFX-feasible. Now, if

- agent 3 finds X_2 EFX-feasible, then we have an EFX allocation where agent 1 gets X_1, agent 2 gets X_3, and agent 3 gets X_2. We can give a symmetric argument when agent 3 finds X_1 EFX-feasible.

- Similarly, if agent 2 finds X_2 EFX-feasible, then we can let agent 3 pick her favorite bundle in the partition (which is obviously EFX-feasible for her) and still give agents 1 and 2 an EFX-feasible bundle. We can give a symmetric argument when agent 2 finds X_1 EFX-feasible.

Therefore, we only need to consider the scenario where only X_3 is EFX-feasible for agents 2 and 3. Essentially, in this scenario, X_3 is "too valuable" to agents 2 and 3, as they do not find any of the remaining bundles EFX-feasible. *A natural attempt would be to remove some good(s) from X_3 and allocate it to X_1 or X_2, i.e., we can increase the valuation of agent 1 for her EFX-feasible bundle(s) by removing the excess goods allocated to the only EFX-feasible bundle of agents 2 and 3.* This brings us to our potential function: $\phi(X) = min(v_1(X_1), v_1(X_2))$. Now, the non-triviality lies in determining the set of goods to be removed from X_3 and then allocating them to X_1 and X_2 such that we maintain our invariants. Although non-trivial, this turns out to be significantly simpler than the procedure used in [27] and also holds when agents 1 and 2 have general monotone valuation functions and agent 3 has an MMS-feasible valuation function.

Before getting into the technicality of the new algorithm, we give the reader a quick recap of the Plaut and Roughgarden (PR) algorithm [59] that determines an EFX allocation when all agents have the same valuation function, $v(\cdot)$ (the only assumption on $v(\cdot)$ is that it is monotone). The algorithm starts with any arbitrary allocation X (which may not be EFX). It makes minor reallocations to improve the valuation of the agent who has the lowest value, i.e., it modifies X to X' such that $min_{i\in[n]}v(X'_i) > min_{i\in[n]}v(X_i)$. We now elaborate on the reallocation procedure: Let ℓ be the agent with the lowest valuation in X. If X is not EFX, then there exists agents i and j such that $v(X_i) < v(X_j \setminus g)$ for some $g \in X_j$. Since $v(X_\ell) \leq v(X_i)$, we also have $v(X_\ell) < v(X_j \setminus g)$. The algorithm removes the good g from j's bundle and allocates it to ℓ. Observe that $v(X_k) > v(X_\ell)$ for all $k \neq \ell$ as we assume non-degeneracy. Also, we have $v(X_\ell \cup g)$ and $v(X_j \setminus g)$ greater than $v(X_\ell)$. Therefore, the valuation of every new bundle is strictly larger than the valuation of X_ℓ. Thus, the valuation of the agent with the lowest valuation improves. This implies that the reallocation procedure will never revisit a particular allocation. As a result, this process will eventually converge to an EFX allocation Y with $v(Y_i) > v(X_\ell)$ for all $i \in [n]$. Formally,

Lemma 6.4 ([59])**.** *Let $X = (X_1, \ldots, X_n)$ be an arbitrary n-partition. Running the PR algorithm with any monotone valuation v results in an EFX-partition $X' = (X'_1, \ldots, X'_n)$ with*

$$\min(v(X_1), \ldots, v(X_n)) \leq \min(v(X'_1), \ldots, v(X'_n)).$$

We have equality only if the input is already EFX for v.

We now elaborate on our algorithm. We give the proof here assuming only monotonicity for the valuation functions of agents 1 and 2 and assuming MMS-feasibility for the valuation of agent 3, i.e., $v_1(\cdot)$ and $v_2(\cdot)$ are general monotone valuation functions and $v_3(\cdot)$ is MMS-feasible. We maintain a partition (X_1, X_2, X_3) of the good set such that

- X_1 and X_2 are EFX-feasible for agent 1.
- X_3 is EFX-feasible for at least one of agents 2 and 3.

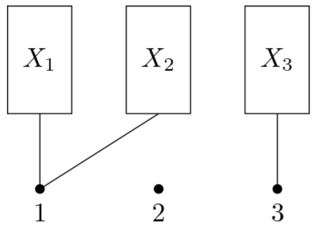


Figure 6.1: The nodes correspond to agents and an edge from agent i to a bundle X_j means that X_j is EFX-feasible for i. In this example, X_1 and X_2 are EFX-feasible for agent 1, and X_3 is EFX-feasible for agent 3. Therefore, the invariants hold.

See Figure 6.1 for better intuition.

One can show the existence of allocations satisfying the above invariants by running the PR algorithm and initializing: Agent 1 runs the PR algorithm with $v = v_1$ to determine a partition (X_1, X_2, X_3) such that all the three bundles are EFX-feasible for her. Then, agent 3 picks her favorite bundle out of the three, say X_3. Clearly, X_3 is EFX-feasible for agent 3, and X_1 and X_2 are EFX-feasible for agent 1. Thus, X satisfies the invariants.

We define our potential function as $\phi(X) = \min(v_1(X_1), v_1(X_2))$. We now elaborate on how to modify X and improve the potential when we cannot determine an EFX allocation from the partition X, i.e., we cannot determine an allocation of the bundles in X to the agents that satisfy the EFX property.

6.2.1 Reallocation When We Cannot Get an EFX Allocation from X

Let $X = (X_1, X_2, X_3)$ be a partition satisfying the invariants. Without loss of generality, let us assume that agent 2 finds X_3 EFX-feasible. Observe that if any one of agents 2 or 3 finds bundles X_1 or X_2 EFX-feasible, then we are done: If agent 3 finds one of X_1 or X_2 EFX-feasible, then we can allocate agent 3's EFX-feasible bundle to her, X_3 to agent 2 and the remaining bundle of X_1 and X_2 to agent 1 and get an EFX allocation. Similarly, if agent 2 finds X_1 or X_2 EFX-feasible, we ask agent 3 to pick her favorite bundle out of X_1, X_2, and X_3. Now, note that no matter which bundle agent 3 picks, there is always a way to allocate agents 1 and 2 their EFX-feasible bundles as agent 1 finds X_1 and X_2 EFX-feasible and agent 2 finds X_3 and at least one of X_1 or X_2 EFX-feasible. If agent 3 picks X_1, allocate X_2 to agent 1 and X_3 to agent 2. If agent 3 picks X_2, allocate X_1 to agent 1 and X_3 to agent 2. Finally, if she picks X_3, allocate the bundle among X_1 and X_2, which is EFX-feasible for agent 2 and the remaining bundle to agent 1. Therefore, from here on we assume that neither agent 2 nor agent 3 finds X_1 or X_2 EFX-feasible. Let g_i be the good in X_3 such that $v_i(X_3 \setminus g_i) \geq v_i(X_3 \setminus h)$ for all $h \in X_3$, i.e., $X_3 \setminus g_i$ is the most valued proper subset of X_3 for agent i.

Observation 6.5. *For $i \in \{2, 3\}$, we have $v_i(X_3 \setminus g_i) > max(v_i(X_1), v_i(X_2))$.*

Proof. We prove for $i = 2$. The proof for $i = 3$ is identical. Let us assume otherwise and say without loss of generality $v_2(X_1) > v_2(X_3 \setminus g_2)$. Then, the only reason why X_1 is not EFX-feasible for agent 2 is if $v_2(X_1) < v_2(X_2 \setminus g)$ for some $g \in X_2$. But, in that case, we have

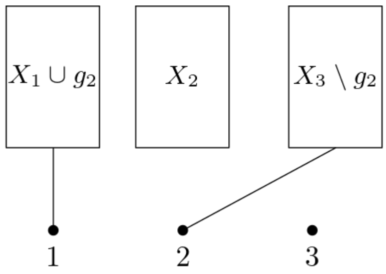


Figure 6.2: Assuming that $v_2(X_3 \setminus g_2) > v_2(X_1 \cup g_2)$, the edge between agent 2 and $X_3 \setminus g_2$ exists. Also, the edge between agent 1 and $X_1 \cup g_2$ exists.

$v_2(X_2) > v_2(X_1) > v_2(X_3 \setminus g_2)$. Therefore, we have $v_2(X_2) > max_{\ell \in [3]} max_{h \in X_\ell} v_2(X_\ell \setminus h)$, implying that X_2 is EFX-feasible for agent 2, which is a contradiction. □

Without loss of generality, assume that $v_1(X_1) < v_1(X_2)$, implying that $\phi(X) = v_1(X_1)$. We now distinguish two cases depending on how valuable the bundle $X_1 \cup g_i$ is to agent i for $i \in \{2, 3\}$ and give the appropriate reallocations in both cases. In particular, we first look into the case where $X_3 \setminus g_i$ is still more valuable to agent i than $X_1 \cup g_i$ for at least one $i \in \{2, 3\}$.

Case: $v_2(X_3 \setminus g_2) > v_2(X_1 \cup g_2)$ or $v_3(X_3 \setminus g_3) > v_3(X_1 \cup g_3)$, i.e., $X_3 \setminus g_i$ is the favourite bundle for agent i in the partition $X_1 \cup g_i$, X_2 and $X_3 \setminus g_i$ for at least one $i \in \{2, 3\}$.

We provide the reallocation rules assuming that $v_2(X_3 \setminus g_2) > v_2(X_1 \cup g_2)$. The rules for the case $v_3(X_3 \setminus g_3) > v_3(X_1 \cup g_3)$ is symmetric. Now, consider the partition $(X_1 \cup g_2, X_2, X_3 \setminus g_2)$ (see Figure 6.2).

By Observation 6.5, $v_2(X_3 \setminus g_2) > v_2(X_2)$ and by our current case $v_2(X_3 \setminus g_2) > v_2(X_1 \cup g_2)$, implying that $X_3 \setminus g_2$ is an EFX-feasible bundle for agent 2. Let X'_1 be a minimal subset of $X_1 \cup g_2$ with respect to set inclusion that agent 1 values more than X_1, i.e., agent 1 values X_1 more than any proper subset of X'_1 and $v_1(X'_1) > v_1(X_1)$. Let $X'_2 = X_2$ and $X'_3 = (X_3 \setminus g_2) \cup ((X_1 \cup g_2) \setminus X'_1)$. We define the partition $X' = (X'_1, X'_2, X'_3)$. Observe that $\phi(X') > \phi(X)$ as $v_1(X'_2) = v_1(X_2) > v_1(X_1)$ (by assumption) and $v_1(X'_1) > v_1(X_1)$ (by definition). Also, note that X'_3 is EFX-feasible for agent 2 as it is the most valuable bundle in X' for agent 2. Now, if X'_1 and X'_2 are EFX-feasible for agent 1, all invariants are maintained, and we are done. So now we look into the case when at least one of X'_1 and X'_2 is not EFX-feasible for agent 1 in X'.

We first make an observation on agent 1's valuation on the bundles X'_1 and X'_2.

Observation 6.6. *We have $v_1(X'_1) > v_1(X'_2 \setminus g)$ for all $g \in X'_2$ and $v_1(X'_2) > v_1(X'_1 \setminus h)$ for all $h \in X'_1$.*

Proof. Note that $v_1(X'_1) > v_1(X_1)$ by definition of X'_1 and $v_1(X_1) > v_1(X_2 \setminus g)$ for all $g \in X_2$ as X_1 was EFX-feasible for agent 1 in X. Since $X'_2 = X_2$, we have $v_1(X'_1) > v_1(X'_2 \setminus g)$ for all $g \in X'_2$.

Similarly, $v_1(X_2) > v_1(X_1)$ by assumption. Furthermore $v_1(X_1) > v_1(X'_1 \setminus h)$ for all $h \in X'_1$ by the definition of X'_1. Since $X'_2 = X_2$, we have $v_1(X'_2) > v_1(X'_1 \setminus h)$ for all $h \in X'_1$. □

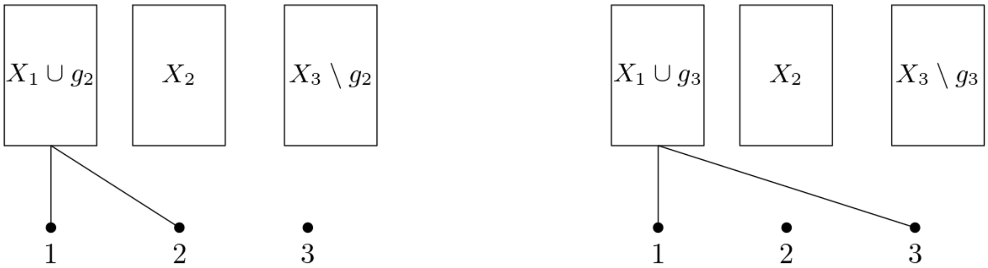


Figure 6.3: Assuming that $v_2(X_3 \setminus g_2) < v_2(X_1 \cup g_2)$ and $v_3(X_3 \setminus g_3) < v_3(X_1 \cup g_3)$, the edge between agent i and $X_1 \cup g_i$ exists in addition to the edge between agent 1 and $X_1 \cup g_i$.

By Observation 6.6, if X'_1 and X'_2 are not EFX-feasible for agent 1 in X', then $v_1(X'_3 \setminus g) > min(v_1(X'_1), v_1(X'_2))$ for some $g \in X'_3$. However, in that case, we run the PR algorithm on the partition X' with agent 1's valuation. Let $Y = (Y_1, Y_2, Y_3)$ be the final partition at the end of the PR algorithm. We have,

$$\begin{aligned} min(v_1(Y_1), v_1(Y_2), v_1(Y_3)) &> min(v_1(X'_1), v_1(X'_2), v_1(X'_3)) \qquad \text{(by Lemma 6.4)} \\ &= min(v_1(X'_1), v_1(X'_2)) \\ &= \phi(X') \\ &> \phi(X). \end{aligned}$$

The equality holds because $v_1(X'_3) > min(v_1(X'_1), v_1(X'_2))$. We then let agent 2 pick her favourite bundle out of Y_1, Y_2, and Y_3. Let us assume without loss of generality that she chooses Y_3. Then, allocation Y satisfies the invariants and we have $\phi(Y) = min(v_1(Y_1), v_1(Y_2)) \geq min(v_1(Y_1), v_1(Y_2), v_1(Y_3)) > \phi(X)$. Thus, we are done.

Remark: Note that we have not used the MMS-feasibility of $v_3(\cdot)$ yet. All the arguments, in this case, hold when all three valuation functions are general monotone. We use MMS-feasibility crucially in the upcoming case.

Case: $v_2(X_3 \setminus g_2) < v_2(X_1 \cup g_2)$ and $v_3(X_3 \setminus g_3) < v_3(X_1 \cup g_3)$, i.e., $X_1 \cup g_i$ is the favourite bundle in the partition $X_1 \cup g_i$, X_2 and $X_3 \setminus g_i$ for all $i \in \{2, 3\}$

(see Figure 6.3). From Observation 6.5, we have $X_3 \setminus g_i >_i X_2$ for $i \in \{2, 3\}$. Therefore, we have,

$$v_2(X_2) < v_2(X_3 \setminus g_2) < v_2(X_1 \cup g_2) \quad \text{and}$$

$$v_3(X_2) < v_3(X_3 \setminus g_3) < v_3(X_1 \cup g_3).$$

By MMS-feasibility of valuation function $v_3(\cdot)$, we conclude that $v_3(X_2) < max(v_3(Z), v_3(Z'))$ where (Z, Z') is any 2-partition of the good set $X_1 \cup X_3$, as MMS-feasibility implies that $max(v_3(Z), v_3(Z')) \geq min(v_3(X_1 \cup g_3), v_3(X_3 \setminus g_3)) > v_3(X_2)$. We run the PR algorithm on the 2-partition $(X_1 \cup g_2, X_3 \setminus g_2)$ with agent 2's valuation $v_2(\cdot)$. Note that this time, we run the PR algorithm with $n = 2$ instead of the usual $n = 3$ in the prior cases. Let (Y_2, Y_3) be the output of the PR algorithm. We let agent 3 choose her favourite among Y_2

and Y_3. Assume without loss of generality, she chooses Y_3. Now, consider the allocation X':

$$\text{agent } 1: \; X_2 \quad \text{agent } 2: \; Y_2 \quad \text{agent } 3: \; Y_3.$$

We now analyze the strong envy in the allocation. To this end, we first observe that agents 2 and 3 do not strongly envy anyone.

Observation 6.7. *Y_2 is EFX-feasible for agent 2 and Y_3 is EFX-feasible for agent 3 in X'.*

Proof. Since (Y_2, Y_3) is the output of the PR algorithm run on $(X_1 \cup g_2, X_3 \setminus g_2)$ with agent 2's valuation function, (i) $v_2(Y_2) > v_2(Y_3 \setminus h)$ for all $h \in Y_3$, and (ii) $v_2(Y_2) \geq min(v_2(X_1 \cup g_2), v_2(X_3 \setminus g_2)) > v_2(X_2)$, where the first inequality follows from Lemma 6.4 and the second inequality follows from the fact that $v_2(X_1 \cup g_2) > v_2(X_3 \setminus g_2) > v_2(X_2)$. Therefore Y_2 is EFX-feasible for agent 2.

Now, we consider agent 3. Note that $Y_3 = max(v_3(Y_2), v_3(Y_3))$ as agent 3 picks her favourite among Y_2 and Y_3. Therefore, $v_3(Y_3) > v_3(Y_2)$ where the strict inequality follows due to non-degeneracy. Furthermore, due to the MMS-feasibility of $v_3(\cdot)$ and the fact that (Y_2, Y_3) is a 2-partition of the good set $X_1 \cup X_3$, we have $v_3(Y_3) = max(v_3(Y_2), v_3(Y_3)) > v_3(X_2)$. Therefore, $v_3(Y_3) > max(v_3(Y_2), v_3(X_2))$ and thus Y_3 is an EFX-feasible bundle for agent 3. □

Therefore, the only possible strong envy is from agent 1. We now enlist the possible strong envy that may arise from agent 1 and show corresponding reallocations.

- Agent 1 does not strongly envy Y_2 and Y_3: Then we are done as X' is an EFX allocation.

- Agent 1 strongly envies both Y_2 and Y_3: In this case, we have $v_1(Y_2) > v_1(X_2)$ and $v_1(Y_3) > v_1(X_2)$. We run the PR algorithm on the partition (X_2, Y_2, Y_3) with agent 1's valuation function $v_1(\cdot)$ and let agent 2 pick her favourite bundle from the final partition X'' returned by the PR algorithm. Then, we have a partition that satisfies the invariants and $\phi(X'') > \phi(X)$ as $min(v_1(X''_1), v_1(X''_2), v_1(X''_3)) > min(v_1(X_2), v_1(Y_2), v_1(Y_3)) = v_1(X_2) > v_1(X_1) = \phi(X)$, where the first inequality follows from Lemma 6.4.

- Agent 1 strongly envies one of Y_2 and Y_3: Let us assume without loss of generality that agent 1 strongly envies Y_2. Let $\overline{Y}_2$ be the minimal subset of Y_2 with respect to set inclusion that agent 1 values more than X_2. Then, consider the partition $X'' = (X''_1, X''_2, X''_3)$ where $X''_1 = X_2$, $X''_2 = \overline{Y}_2$ and $X''_3 = Y_3 \cup (Y_2 \setminus \overline{Y}_2)$. First note that X''_3 is EFX-feasible for agent 3 as $X'_3 = Y_3$ was EFX-feasible in allocation X' and now the bundle X''_1 remains the same, the bundle X'_2 has been compressed further in X'', and $X'_3 \subset X''_3$. Also note that $\phi(X'') = min(v_1(X''_1), v_1(X''_2)) = min(v_1(X_2), v_1(\overline{Y}_2)) = v_1(X_2) > v_1(X_1) = \phi(X)$. If X''_1 and X''_2 are EFX-feasible for agent 1, then partition X'' satisfies the invariants and $\phi(X'') > \phi(X)$ and we are done. So now consider the case when at least one of X''_1 and X''_2 is not EFX-feasible for agent 1. Note that $v_1(X''_1) > v_1(X''_2 \setminus h)$ for all $h \in X''_2$ and $v_1(X''_2) > v_1(X''_1)$ by the fact that $X''_1 = X_2$ and by the definition of $X''_2 = \overline{Y}_2$. Thus, if one of X''_1 or X''_2 is not EFX-feasible for agent 1, then we must have

$v_1(X_3'' \setminus h') > min(v_1(X_1''), v_1(X_2''))$ for some $h' \in X_3''$. In this case, we run the PR algorithm on the partition (X_1'', X_2'', X_3'') with agent 1's valuation function $v_1(\cdot)$ and let agent 2 pick her favourite bundle from the final partition Z returned by the PR algorithm. Then Z satisfies the invariants and

$$\begin{aligned}\phi(Z) &\geq min(v_1(Z_1), v_1(Z_2), v_1(Z_3)) \\ &\geq min(v_1(X_1''), v_1(X_2''), v_1(X_3'')) \\ &= v_1(X_2) \\ &> v_1(X_1) \\ &= \phi(X).\end{aligned}$$

So we are done.

This brings us to the main result of this section.

Theorem 6.2. *EFX allocations exist with three agents as long as at least one agent has an MMS-feasible valuation function.*

CHAPTER 7

Approximate EFX with Bounded Charity

Caragiannis et al. [23] introduced the notion of EFX with charity. The goal here is to find "good" partial EFX allocations, i.e., partial EFX allocations where the set of unallocated goods is not very valuable. Following the same line of work, Chaudhury et al. [29] showed the existence of a partial EFX allocation X such that no agent envies the set of unallocated goods and the total number of unallocated goods is at most $n-1$. Later, Chaudhury et al. [28] gave a polynomial time algorithm to compute a $(1-\varepsilon)$-EFX allocation with $\mathcal{O}((n/\varepsilon)^{4/5})$ charity for any $\varepsilon > 0$. One key aspect of the technique in [28] is the reduction of the problem of improving the bounds on charity to a purely graph-theoretic problem. In particular, [28] defines the notion of a *rainbow cycle number*: Given an integer $d > 0$, the rainbow cycle number $R(d)$ is the largest k such that there exists a k-partite graph $G = (V_1 \cup V_2 \cup \cdots \cup V_k, E)$ such that

- each part has at least 1 and at most d vertices, i.e., $1 \leq |V_i| \leq d$, and
- every vertex in G has exactly one incoming edge from every part in G except the part containing it, and
- there exists no cycle C in G that visits each part at most once. That is, every cycle C in G must have $|V(C) \cap V_i| > 1$ for some $V_i, i \in [k]$, where $V(C)$ is the set of vertices of cycle C.

Let $h^{-1}(d)$ denote the smallest integer ℓ such that $h(\ell) = \ell \cdot R(\ell) \geq d$. Then there always exist an $(1-\varepsilon)$-EFX allocation with $\mathcal{O}(\frac{n}{\varepsilon \cdot h^{-1}(n/\varepsilon)})$ charity. So, the smaller the upper bound on $h(\ell)$, the lower the number of unallocated goods. [28] shows that $R(d) \in \mathcal{O}(d^4)$ and thus establish the existence of $(1-\varepsilon)$-EFX allocation with $\mathcal{O}((n/\varepsilon)^{4/5})$ charity. An upper bound of $\mathcal{O}(d^2 2^{(\log\log d)^2})$ was obtained by [14], thereby showing the existence of EFX allocations with $\mathcal{O}((n/\varepsilon)^{0.67})$ charity. In this paper, we close this line of improvements by proving an almost tight upper bound on d (matching the lower bound up to a log factor). We note that our technique (and analysis) is simpler than the ones used for upper-bounding rainbow cycle number in [14, 28, 49]. Although the core argument is based on the *probabilistic method*, we also derandomize the approach to construct such an allocation in deterministic polynomial time.

Theorem 7.1. *Given any integer $d > 0$, the rainbow cycle number $R(d) \in \mathcal{O}(d \log d)$.*

For any allocation X, let us denote the Nash welfare of X by $NW(X)$. As a consequence of the improved bound in Theorem 7.1, we obtain:

Theorem 7.2. *There exists a deterministic polynomial time algorithm that determines a partial $(1-\varepsilon)$-EFX allocation X such that no agent envies the set of unallocated* *goods and the total number of unallocated goods is* $\tilde{\mathcal{O}}((n/\varepsilon)^{1/2})$[1]. *Furthermore,* $NW(X) \geq 1/(2e^{1/e}) \cdot NW(X^*)$ *where* X^* *is the allocation with maximum Nash welfare.*

Rainbow Cycle and Zero-sum Combinatorics. We believe that investigating tighter bounds on $R(d)$ is interesting in its own right. [14] showed intriguing connections between rainbow cycle number and zero-sum problems in extremal combinatorics. Zero-sum problems in graphs ask questions of the following flavor: Given an edge/vertex weighted graph, whether there exists a certain substructure (for example, cliques, cycles, paths, etc.) with a zero-sum (modulo some integer). In particular, [14] shows that the rainbow cycle number is a natural generalization of the zero-sum problems studied by [2] and [57]. Both papers [2, 57] aim to upper bound the maximum number of vertices of a complete bidirected graph with integer edge labels, avoiding a zero-sum cycle (modulo d). [14] shows through a simple argument that this is upper bounded by the *permutation rainbow cycle number* $R_p(d)$, which is defined by introducing an additional constraint in the definition of $R(d)$ that for all i, j, each vertex in V_i has exactly one *outgoing* edge to some vertex in V_j (in addition to exactly one incoming edge from some vertex in V_j). In Section 7.2.2, we show through a simple argument that $R_p(d) \leq 2d - 2$, thereby also improving the upper bounds of $\mathcal{O}(d \log(d))$ in [2] and $8d - 1$ in [57].

Lemma 7.3. *For* $d > 1$*, we have* $R_p(d) \leq 2d - 2$*. This yields an alternate proof for the key lemma in [14] that the maximum number of vertices of a complete bidirected graph with integer edge labels avoiding a zero-sum cycle (modulo* d*) is at most* $2d - 2$*.*

Independent Work.

Independently and concurrently to our work, [49] also investigated upper bounds on the rainbow cycle number, and they also showed $R(d) \in \mathcal{O}(d \log(d))$. Unlike our approach that utilizes probabilistic methods, [49] introduced another problem in extremal combinatorics called *rainbow path degree*. They showed a relation between rainbow path degree $H(\ell)$ and rainbow cycle number $R(d)$. Then by proving $H(\ell) = \Omega(\ell^2/\log \ell)$, they showed $R(d) = \mathcal{O}(d \log d)$. We would like to emphasize that our techniques are entirely different and our proof is significantly shorter and simpler.

In another independent work, [14] established the connections between rainbow cycle number and problems in zero sum extremal combinatorics and obtained the same upper bound of $2d - 2$ for $R_p(d)$. This bound was later improved to $2d - 4$ for large enough d by [49].

7.1 Notations and Tools

[28] reduced the problem of finding approximate EFX allocations with sublinear charity to a problem in extremal graph theory. In particular, they introduced the notion of a rainbow cycle number.

Definition 7.4. *Given an integer* $d > 0$*, the rainbow cycle number* $R(d)$ *is the largest* k *such that there exists a* k*-partite directed graph* $G = (V_1 \cup V_2 \cup \cdots \cup V_k, E)$ *such that*

[1] $\tilde{\mathcal{O}}$ ignores logarithmic factors.

- *each part has at least* 1 *and at most* d *vertices, i.e.,* $1 \leq |V_i| \leq d$, *and*
- *every vertex has exactly one incoming edge from every part other than the one containing it*[2], *and*
- *there exists no cycle* C *in* G *that visits each part at most once. That is, every cycle* C *in* G *must have* $|V(C) \cap V_i| > 1$ *for some* $V_i, i \in [k]$, *where* $V(C)$ *is the set of vertices of cycle* C.

We also refer to cycles that visit each part at most once as "rainbow" cycles.

They show that any finite upper bound on $R(d)$ implies the existence of approximate EFX allocations with sublinear charity. Better upper bounds on $R(d)$ give better bounds on the charity. In particular, they prove the following theorem.

Theorem 7.5 ([28]). *Let* $G = (V_1 \cup V_2 \cup \ldots V_k, E)$ *be a* k*-partite digraph such that (i) each part has at most* d *vertices and (ii) each vertex in* G *has an incoming edge from every part other than the one containing it. Furthermore, let* $T(d)$ *be such that* $k > T(d) \geq R(d)$. *If there exists a polynomial time algorithm that can find a cycle visiting each part at most once in* G, *then there exists a polynomial time algorithm that determines a partial EFX allocation* X *such that*

- *the total number of unallocated goods is in* $\mathcal{O}(n/\varepsilon \cdot h^{-1}(n/\varepsilon))$ *where* $h^{-1}(d)$ *is the smallest integer* ℓ *such that* $h(\ell) = \ell \cdot T(\ell) \geq d$.
- $NW(X) \geq 1/(2e^{1/e}) \cdot NW(X^*)$, *where* X^* *is the allocation with maximum Nash welfare.*

7.2 Bounds on Rainbow Cycle Number

In this section, we improve the upper bounds on the rainbow cycle number introduced in [28], thereby implying the existence of approximate EFX allocations with $\tilde{\mathcal{O}}((n/\varepsilon)^{1/2})$ charity. [28] gave an upper bound of $R(d) \in \mathcal{O}(d^4)$ and they showed it results in the existence of a $(1-\varepsilon)$-EFX allocation with $\mathcal{O}((n/\varepsilon)^{4/5})$ charity. In the same paper, [28] shows a lower bound of d on $R(d)$. In this section, we show improved bounds on $R(d)$. In particular, in Section 7.2.1, we show that $R(d) \in \mathcal{O}(d \log d)$ (making the upper bound almost tight), thereby implying the existence of $(1-\varepsilon)$-EFX allocations with $\tilde{\mathcal{O}}((n/\varepsilon)^{1/2})$ charity.

Our technique to achieve the improved bound involves the probabilistic method. It is significantly simpler and yields better guarantees. We briefly sketch our algorithmic proof. Let there be k parts in $G = (V_1 \cup V_2 \cup \ldots V_k, E)$. Note that each part has at most d vertices, and each vertex has exactly one incoming edge from every part. We pick one vertex v_i from each part V_i uniformly and independently at random. Now, it suffices to show that with non-zero probability, the induced graph on the vertices $v_1, v_2, \ldots, v_k$ is cyclic for some $k \in \mathcal{O}(d \log d)$. Note that if every vertex in $G[v_1, \ldots, v_k]$

[2] In the original definition of the rainbow cycle number $R(d)$ in [28], every vertex can have more than one incoming edge from a part. However, by reducing the number of edges, we can only increase the upper bound on $R(d)$.

has an incoming edge, then $G[v_1 \dots v_k]$ is cyclic. So we need to show a non-zero lower bound on the probability of each vertex having at least one incoming edge or equivalently show an upper bound on the probability that each vertex has no incoming edge in $G[v_1 \dots v_k]$. To this end, let E_{v_i} denote the event that vertex v_i has no incoming edge in $G[v_1 \dots v_k]$. Note that $\mathbf{P}[E_{v_i}] \leq (1 - 1/d)^{k-1}$: v_i has at least one incoming edge from each part, and therefore, the probability that there is no incoming edge from v_j to v_i is at most $(1 - 1/d)$ for all j. Since all v_j's are independently chosen, the probability that v_i has no incoming edge from any part is at most $(1 - 1/d)^{(k-1)}$. Then, by union bound, $\mathbf{P}[\cup_{i \in [n]} E_{v_i}] \leq \sum_{i \in [n]} \mathbf{P}[E_{v_i}] \leq k(1-1/d)^{(k-1)}$. Therefore, the probability that $G[v_1 \dots v_k]$ is cyclic is at least $1 - k(1 - 1/d)^{(k-1)}$ which is strictly positive for $k \in \mathcal{O}(d \log d)$.

In Section 7.2.2, we show an upper bound of $2d - 2$ assuming that every vertex $v \in V_i$ has exactly one incoming edge from any other part $V_j \neq V_i$ and exactly one outgoing edge to some vertex in V_j. We call this number $R_p(d)$. We remark that the lower bound of d in [28] also holds for $R_p(d)$. The upper bound of $2d - 2$ immediately improves the upper bound on the zero-sum extremal problem studied in [2, 57].

7.2.1 Almost Tight Upper Bound on $R(d)$

Recall that $R(d)$ is the largest k such that there exists a k-partite digraph G with k classes of vertices V_i so that each part V_i has at most d vertices for all distinct i, j each vertex in V_i has an incoming edge from some vertex in V_j, and there exists no (directed) rainbow cycle, namely, a cycle in G that contains at most one vertex of each V_i. In this section, we prove the following improved bound which is tight up to the logarithmic factor.

Theorem 7.6. *If*

$$k(1 - {}^1\!/\!_d)^{k-1} < 1 \tag{7.1}$$

then $R(d) < k$. Therefore, $R(d) = \mathcal{O}(d \log d)$.

Proof. Suppose $k(1 - {}^1\!/\!_d)^{k-1} < 1$. Let S be a random set of k vertices of G obtained by picking a single vertex v_i in each V_i, randomly and uniformly among all vertices of V_i, where all choices are independent. For each vertex v, let E_v be the event that S contains v and contains no other vertex u so that uv is a directed edge. We claim that if $v \in V_i$ then the probability of E_v is at most

$$\frac{1}{|V_i|}(1 - {}^1\!/\!_d)^{k-1}.$$

Indeed, the probability that $v \in S$ is $1/|V_i|$. Conditioning on that, since for every $j \neq i$ there is some $u_j \in V_j$ so that $u_j v$ is a directed edge, and the probability that u_j is in S is $1/|V_j| \geq 1/d$, the probability that v has no in-neighbor in V_j is at most $1 - 1/d$. As the choices are independent, the claim follows. By the union bound, the probability that there is a vertex v so that the event E_v occurs is at most

$$\sum_{i=1}^{k} |V_i| \frac{1}{|V_i|}(1 - {}^1\!/\!_d)^{k-1} = k(1 - 1/d)^{k-1} < 1.$$

Therefore, with positive probability, every vertex in the induced subgraph of G on S has an in-neighbor. Hence, there is such an S and in this induced subgraph, there is a cycle, which contains at most one vertex from each V_i. Thus $R(d) < k$.

Setting $k = 2d \ln d + 1$, we have

$$\begin{aligned} k(1-\frac{1}{d})^{k-1} &\le ke^{-\frac{1}{d}(k-1)} && (1+x \le e^x \text{ for all real } x) \\ &= ke^{-2\ln d} && (k = 2d\ln d + 1) \\ &= (2d\ln d + 1)(\frac{1}{d})^2 \\ &< 1. && (\text{for large enough } d) \end{aligned}$$

Therefore, $R(d) = \mathcal{O}(d \log d)$. □

Theorem 7.5 and Theorem 7.6 then imply Theorem 7.2.

Remark. The proof above can be derandomized using the method of conditional expectations (cf., e.g., [3], Chapter 16), giving the following.

Proposition 7.7. *Let G be a k-partite digraph with classes of vertices V_i, each having at most d vertices. Suppose that for all distinct i, j, each vertex in V_i has an incoming edge from some vertex in V_j and vice versa, and suppose that (7.1) holds. Then a rainbow cycle in G can be found by a deterministic polynomial time algorithm.*

Proof. We apply the method of conditional expectations to produce a set $S = \{s_1, s_2, \ldots, s_k\}$ of vertices of G, where $s_i \in V_i$, so that every indegree in the induced subgraph of G on S is positive. This is done by choosing the vertices s_i one by one in order, maintaining a potential function ϕ whose value is the conditional expectation of the number of events E_v that hold, given the choices of the vertices s_i made so far.

In the beginning, there are no choices made, and the value of ϕ is the sum

$$\sum_{i=1}^{k} |V_i| \frac{1}{|V_i|}(1 - 1/d)^{k-1} = k(1 - 1/d)^{k-1} < 1.$$

Assuming $s_1, s_2, \ldots, s_{i-1}$ have already been chosen, and the above conditional expectation is still smaller than 1, choose $s_i \in V_i$ as the vertex that minimizes the updated value of the conditional expectation. As the expectation is the average over all possible choices of s_i, this minimum stays below 1. The computation of the required conditional expectations for each of the possible $|V_i| \le d$ choices of $s_i \in V_i$ can be done efficiently. At the end of the process, the value of the potential function is exactly the number of events E_v that hold, and since this number is below 1, none of them holds. This supplies the required set S. Starting in any vertex of S and moving repeatedly to an in-neighbor of it in S until we reach a vertex we have already visited supplies the desired rainbow cycle. □

7.2.2 A Linear Upper Bound on $R_p(d)$

In this section, we assume graph G satisfies all the properties in Definition 7.4 and also for all different parts V_i and V_j, each vertex in V_i has exactly one outgoing edge to a vertex in V_j. We call these graphs permutation graphs since the set of edges from any part to any other part defines a permutation.

Definition 7.8. *Given an integer $d > 0$, the permutation rainbow cycle number $R_p(d)$ is the largest k such that there exists a k-partite directed graph $G = (V_1 \cup V_2 \cup \cdots \cup V_k, E)$ such that*

- *each part has exactly d vertices, i.e., $|V_i| = d$, and*
- *every vertex has exactly one incoming edge from every part other than the one containing it, and*
- *every vertex has exactly one outgoing edge to every part other than the one containing it, and*
- *there exists no cycle C in G that visits each part at most once. That is, every cycle C in G must have $|V(C) \cap V_i| > 1$ for some $V_i, i \in [k]$, where $V(C)$ is the set of vertices of cycle C.*

Theorem 7.9. *For all integers $d > 1$, $R_p(d) < 2d - 1$.*

In the rest of this section, we prove Theorem 7.9. The proof is by induction.

Basis: For the base case, consider $d = 2$. For the sake of contradiction assume $R(2) \geq 3$ and let $V_1 = \{a_1, a_2\}$, $V_2 = \{b_1, b_2\}$ and $V_3 = \{c_1, c_2\}$ be three different parts. Without loss of generality, we can assume there is an edge from a_1 to b_1 and one from b_1 to c_1. Assuming there is no cycle in this graph, a directed edge from c_1 to a_1 cannot exist, and therefore, a directed edge from c_2 to a_1 exists. Thus, no edge from a_1 to c_2 exists, implying an edge from a_1 to c_1. Also, since there is an edge from b_1 to c_1, there must be an edge from c_1 to b_2 (since there can be none to b_1). Now if there is an edge from b_1 to a_1, the cycle (a_1, b_1) exists, and if there is an edge from b_2 to a_1, the cycle (a_1, c_1, b_2) exists which is a contradiction. Therefore, $R(2) < 3$.

Moreover, we prove that $R_p(1) = 1$. Towards a contradiction assume $R_p(1) \geq 2$ and there are two different parts $V_1 = \{a\}$ and $V_2 = \{b\}$. Then, there exists an edge from a to b and one from b to a, forming a cycle.

Induction step: For $d > 2$, we assume for all $1 < d' < d$, $R_p(d') < 2d' - 1$ and prove $R_p(d) < 2d - 1$. In particular, since $R_p(1) = 1$, for all $1 \leq d' < d$ we have

$$R_p(d') \leq 2d' - 1. \tag{7.2}$$

First, we define i-restricted paths, which are the paths that use each part at most once, and except for the last vertex, all vertices are in the first i parts.

Definition 7.10. *We call path $P = v_1 \to v_2 \to \cdots \to v_t$ an i-restricted path if*

- *$v_1, \ldots, v_{t-1} \in V_1 \cup V_2 \cup \cdots \cup V_i$, and*
- *P visits each part at most once.*

Note that for all $j > i$, every i-restricted path is also a j-restricted path. Now we prove the following lemma.

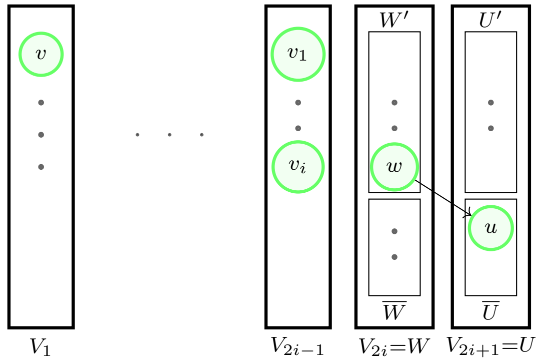


Figure 7.1: W' has an outgoing edge to $\overline{U}$

Lemma 7.11. *If for all $1 \leq d' < d$, $R_p(d') \leq 2d' - 1$, then the following holds. For all $k \geq 2d - 1$ and for every vertex v, there is a way of reindexing the parts such that $v \in V_1$ and for all $i \in [d]$, there are i nodes in V_{2i-1} which are reachable from v via $(2i-2)$-restricted paths.*

Proof. The proof of the claim is also by induction. Let v be a fixed vertex. For the base case, let $i = 1$. If $v \in U$, set $V_1 = U$, and the claim follows.

For the induction step from i to $i+1$, we assume $V_1, V_2, \dots, V_{2i-1}$ are already defined and for all $j \in [i]$, there is a $(2i-2)$-restricted path from v to $v_j \in V_{2i-1}$. For all parts $Z \notin \{V_1, V_2, \dots, V_{2i-1}\}$ and all $j \in [i]$, let $v_j \to z_j$ be the outgoing edge from v_j to Z. Since each node in V_{2i-1} has exactly one outgoing edge to Z and each node in Z has exactly one incoming edge from V_{2i-1}, the nodes $z_1, z_2, \dots, z_i$ are distinct. Therefore, for all parts $Z \notin \{V_1, V_2, \dots, V_{2i-1}\}$, at least i nodes in Z are reachable from v via $(2i-1)$-restricted paths. For all parts $Z \notin \{V_1, V_2, \dots, V_{2i-1}\}$, let $Z' \subseteq Z$ be the vertices that are reachable from v via $(2i-1)$-restricted paths and let $\overline{Z} = Z \setminus Z'$. If there exists a part $U \notin \{V_1, V_2, \dots, V_{2i-1}\}$ such that $|U'| \geq i+1$, we set $V_{2i} = W$ for some $W \notin \{V_1, V_2, \dots, V_{2i-1}, U\}$ and set $V_{2i+1} = U$ and the claim follows. Otherwise, for all parts $Z \notin \{V_1, V_2, \dots, V_{2i-1}\}$, we have $|Z'| = i$ and $|\overline{Z}| = d - i$. If there exist $U, W \notin \{V_1, V_2, \dots, V_{2i-1}\}$ such that $w \in W'$ has an outgoing edge to $u \in \overline{U}$, then we set $V_{2i} = W$ and $V_{2i+1} = U$. Note that all nodes in U' are reachable from v using $(2i-1)$-restricted paths, and u is reachable via a $(2i)$-restricted path. Therefore, in total $i+1$ vertices in $U = V_{2i+1}$ are reachable from v via $(2i)$-restricted paths (see Figure 7.1 for an illustration).

Let $V(G) = V_1 \cup V_2 \cup \dots \cup V_{2i-1} \cup U_1 \cup U_2 \cup \dots \cup U_{k-2i+1}$. The only remaining case is that for all $j \in [k-2i+1]$, $|\overline{U}_j| = d - i$ and for all $j, \ell \in [k-2i+1]$, there is no edge from U'_j to $\overline{U}_\ell$. This means that all the $d-i$ incoming edges of $\overline{U}_\ell$ come from $\overline{U}_j$. Hence all the $d-i$ outgoing edges of $\overline{U}_j$ go to $\overline{U}_\ell$. Therefore, the induced subgraph on $\overline{U}_1 \cup \overline{U}_2 \cup \dots \cup \overline{U}_{k-2i+1}$, forms a permutation graph (see Figure 7.2). By the assumption of the lemma, we know $R_p(d-i) \leq 2d - 2i - 1$ and hence, $k - 2i + 1 \leq 2d - 2i - 1$. This contradicts the assumption of the claim, which requires $k \geq 2d - 1$. Therefore, this case cannot occur. □

Back to the assumption step, we want to prove $R_p(d) < 2d-1$. Towards a contradiction, assume $R_p(d) \geq 2d - 1$ and consider a graph G with $|R_p(d)|$ parts satisfying properties

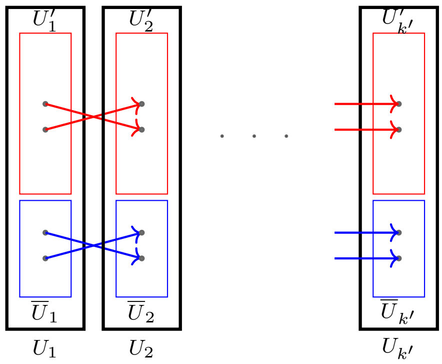


Figure 7.2: $k' \geq k - 2i - 1$ and for all $j, \ell \in [k']$, there exists no edge between U'_j and $\overline{U}_\ell$.

of Definition 7.8. Now, pick an arbitrary vertex v. By setting $k = d$ in Lemma 7.11, there exists a reindexing of the parts such that all d nodes in part V_{2d-1} are reachable from v using $(2d-2)$-restricted paths. Let $u \in V_{2d-1}$ be the vertex with an outgoing edge to v. Then a $(2d-2)$-restricted path from v to u followed by the edge $u \to v$ forms a rainbow cycle. Hence, $R_p(d) < 2d - 1$.

CHAPTER 8

Epistemic EFX for Monotone Valuations

A recent work of Caragiannis et al. [22] introduced a promising relaxation of EFX, called *epistemic* EFX (which adapts the concepts of *epistemic envy-freeness* defined by [9]). We call an allocation X EEFX if for every agent $i \in [n]$, there exists an allocation Y such that $Y_i = X_i$ and for every bundle $Y_j \in Y$, we have $v_i(X_i) \geq v_i(Y_j \setminus \{g\})$ for every $g \in Y_j$. That is, an allocation is EEFX if, for every agent, it is possible to shuffle the items in the remaining bundles so that she becomes "EFX-satisfied". See Example 8.1 for a better intuition.

Example 8.1. *Consider a fair division instance consisting of* 7 *items and* 3 *agents with additive valuations as described in Table 8.1. Now consider the allocation* X *where* $X_1 = \{g_1, g_2, g_4\}, X_2 = \{g_3, g_5, g_6\}$, *and* $X_3 = \{g_7\}$. *Note that* X *is envy-free, and hence,* EFX *and* EEFX. *Now assume that agent* 1 *and* 2 *exchange the items* g_3 *and* g_4. *Formally, let* $Y = (\{g_1, g_2, g_3\}, \{g_4, g_5, g_6\}, \{g_7\})$. *For* $i \in \{1, 2\}$, *have* $v_i(Y_i) = 300 > 201 = v_i(X_i)$, *and* $v_3(Y_3) = v_3(X_3)$. *Therefore, intuitively it seems that* Y *is a better allocation compared to* X *since agents* 1 *and* 2 *are strictly better off and agent* 3 *is as happy as before (i.e.,* Y *Pareto dominates* X*). However, note that while allocation* Y *is still* EEFX, *it is not* EFX. *Namely, agent* 3 *strongly envies agent* 1*:* $v_3(Y_1 \setminus \{g_1\}) = 100 > 55 = v_3(Y_3)$.

Caragiannis et al. [22] establish existence and polynomial-time computability of EEFX allocations for an arbitrary number of agents with a restricted class of *additive* valuations. Thus, the following question naturally arises:

Do EEFX allocations exist for an arbitrary number of agents with general *monotone* valuations?

We answer the above question affirmatively and establish computational hardness and information-theoretic lower bounds for finding EEFX allocations:

(1) EEFX allocations are guaranteed to *exist* for any fair division instance with an arbitrary number of agents having general *monotone* valuations; see Theorem 8.7.

	g_1	g_2	g_3	g_4	g_5	g_6	g_7
v_1	100	100	100	1	1	1	1
v_2	1	1	1	100	100	100	1
v_3	1	50	50	1	1	1	55

Table 8.1: The additive valuation functions of 3 agents for 7 goods.

(2) An exponential number of valuation queries is required by any deterministic algorithm to compute an EEFX allocation for fair division instances with an arbitrary number of agents with identical submodular valuations; see Theorem 8.14.

(3) The problem of computing EEFX allocations for fair division instances with an arbitrary number of agents having identical submodular valuations is PLS-hard; see Theorem 8.15.

It is relevant to note that, with the above results, the notion of *epsitemic*-EFX becomes the *second* known relaxation of EFX (besides EF1), that admits such strong existential guarantees. Along with its hardness results, the notion of EEFX for discrete settings seems to enjoy results of *similar* flavor as that of envy-freeness for cake division [31, 65, 66].

Similar computational hardness and information-theoretic lower bounds are known for computing an EFX allocation between agents with identical submodular valuations, even when $n = 2$; see [59] and [45]. Observe that, the set of EEFX and EFX allocations are identical for two agents. Hence, the computational hardness and information-theoretic lower bounds known for computing EFX allocations between two agents carry forward to EEFX allocations as well, but *only* for two agents. At first sight, it might seem trivial that finding an EEFX allocation can only get harder when the number of agents grows. However, note that when the number of agents grows, more bundles become "EEFX-feasible" for each agent, and hence, finding an EEFX allocation may be done faster. Nevertheless, in this chapter, we prove similar lower bounds for EEFX by reducing the problem of computing an EEFX allocation among an arbitrary number of agents with identical submodular valuations from the problem of computing an EFX allocation among two agents with identical submodular valuations.

We use a similar reduction that was previously used to reduce the problem of finding an EFX allocation for $n > 2$ agents to finding an EFX allocation for 2 agents [45]: take an instance with two identical agents and construct an instance with identical $n > 2$ agents by introducing $n-2$ *large* items. In any EFX allocation, these large items must be allocated to $n-2$ different agents who are not allocated any other item. Hence the problem reduces to finding an EFX allocation of the original items to two agents with identical submodular valuations (the original hard instance).

Unlike EFX, in EEFX allocations it is not necessary for $n-2$ agents to receive only one large item. Consider the following example. $\mathcal{M} = \{g_1, g_2, \ldots, g_{n+1}\}$ and for all agents i, $v_i(g_1) = v_i(g_2) = v_i(g_3) = 1$ and for all $j > 3$, $v_i(g_j) = 100$. In any EFX allocation, two of g_1, g_2, g_3 are allocated to one agent, the remaining item among these three to another agent, and each of the remaining items is allocated to a distinct remaining agent. However, the allocation $(\{g_1\}, \{g_2\}, \ldots, \{g_{n-1}\}, \{g_n, g_{n+1}\})$ which allocates all the first $n-1$ items to $n-1$ different agents and the last two items (with total value 200) to the last agent is EEFX. Therefore, the set of all EFX allocations is a strict subset of the set of all EEFX allocations in some instance with identical additive (therefore also submodular) valuations that could be generated in this reduction. Also, as shown in the example, by introducing $n-2$ large items, it is not necessary that the original items would be allocated to only two agents. This shows that the hardness of finding EEFX allocations for n agents with identical submodular valuations is not an immediate corollary of the hardness of finding EFX (or EEFX) allocations for n (respectively two) agents with identical submodular valuations. However, we use very similar ideas to prove the former.

See Section 8.1 for further discussion on the PLS class [50].

Although similar computational hardness and information-theoretic bounds hold true for finding EFX and EEFX allocations, our work has proved guaranteed existence of EEFX allocations for an arbitrary number of agents with monotone valuations, whereas existence of EFX allocations for more than three agents even with additive valuations remains a major open problem.

8.0.1 Our Techniques

Consider a fair division instance $\mathcal{I} = (\mathcal{N}, \mathcal{M}, \mathcal{V})$ and a *desirable property* $\mathcal{P}$ of a bundle $B \subseteq \mathcal{M}$ for an agent $i \in \mathcal{N}$. For example, in this chapter, we consider the fairness property of whether B is n-*epistemic*-EFX for an agent i (see Definition 2.25). We say B is *desirable* to i when B satisfies the property $\mathcal{P}$ for agent i. The goal is to find an allocation $A = (A_1, \ldots, A_n)$ such that A_i is desirable to each agent $i \in \mathcal{N}$; we call such an allocation *desirable*.

For any partitioning of the items into n bundles $X_1, X_2, \ldots, X_n$, let us consider a bipartite graph $G(X)$ with one side representing the n agents and the other side representing the n bundles. There exists an edge (i, j) between (the node corresponding to) agent i and (the node corresponding to) bundle X_j, if and only if, bundle X_j is *desirable* to agent i. For any subset of the nodes $S \subseteq \mathcal{N}$, let us write $N(S)$ to denote the set of all neighbours of S in $G(X)$.

Note that, if $G(X)$ has a perfect matching, then this matching translates to a *desirable* allocation in $\mathcal{I}$. Therefore, let us assume that $G(X)$ does not admit a perfect matching and hence admits a Hall's violator set. That is, there exists a subset of agents $\{a_1, \ldots, a_{t+1}\}$ for which $N(\{a_1, \ldots, a_{t+1}\}) \le t$. But also, there exists a subset of bundles $\{X_{j_1}, \ldots, X_{j_{k+1}}\}$ for which $N(\{X_{j_1}, \ldots, X_{j_{k+1}}\}) \le k$. Let us assume that $\{X_{j_1}, \ldots, X_{j_{k+1}}\}$ is minimal. If $k \ge 1$, this means that we can find a matching of $((i_1, X_{j_1}), \ldots, (i_k, X_{j_k}))$ such that there exists no edge between agent $i \in \mathcal{N} \setminus \{i_1, \ldots, i_k\}$ and bundles $X_{j_1}, \ldots, X_{j_{k+1}}$. In other words, for all $\ell \in [k]$, X_{j_ℓ} is desirable to i_ℓ and is not desirable to any $i \notin \{i_1, \ldots, i_k\}$.

After finding such a matching, it is intuitive to allocate X_{j_ℓ} to i_ℓ for all $\ell \in [k]$ and then recursively find a desired allocation of the remaining goods to the remaining agents. In order to do so, we need to ensure two important conditions.

(1) We can find a non-empty matching $((i_1, X_{j_1}), \ldots, (i_k, X_{j_k}))$ in each step.

(2) After removing $\{X_{j_1}, \ldots, X_{j_k}\}$ from $\mathcal{M}$, we can still find desirable bundles (with respect to the original instance) for the remaining agents.

Whether ensuring these conditions is possible or not, depends on the property $\mathcal{P}$. In this chapter, we prove this approach works when the property $\mathcal{P}$ is n-epistemic-EFX, and thereby proving the existence of EEFX allocations for monotone valuations.

Although these two conditions might seem inconsequential, we prove that a stronger condition can simultaneously imply both of them. Namely, we only need to prove that at each step with n' remaining agents, for any remaining agent i, we can partition the remaining items into n' many bundles $X_1, \ldots, X_{n'}$ such that X_j is desirable to i for all $j \in [n']$. This way, at each step, we can ask one of the remaining agents to partition the remaining goods into n' many desirable bundles with respect to her own valuation. Then, we either find a perfect matching, or we find a non-empty matching and reduce the size of the instance.

This technique works when the desirable property is, for instance, *proportionality* (moving-knife procedure [33]) or approximate *maximin share* [1, 4, 48, 51, 64]. In this chapter, we show that EEFX allocations under monotone valuations are also compatible with the above technique. Recently, [19] proved this technique also works for finding PROP1[1] allocations among agents with additive valuations in a *comparison-based model*. Here, two bundles are presented to an agent and she responds by telling which bundle she prefers.

8.1 Notations and Tools

For all $i \in \mathcal{N}$, we assume v_i is *monotone*; i.e., for all $i \in \mathcal{N}$, $g \in \mathcal{M}$ and $S \subset \mathcal{M}$, $v_i(S \cup g) \geq v_i(S)$.

For a fair division instance $\mathcal{I} = (\mathcal{N}, \mathcal{M}, \mathcal{V})$ with monotone valuations, we consider the valuations to be accessed via an oracle. Note that, monotone valuations are the most general class of valuations when the set of items consists of only goods or only chores.

Next, we define a notion of EEFX-*graph* that plays a crucial role in proving the existence of EEFX allocations.

Definition 8.2. *For a fair division instance, consider a partition of $\mathcal{M}$ into n bundles $Y_1, \ldots, Y_n$. We define the* EEFX-graph *as an undirected bipartite graph $G = (V, E)$, where V has one part consisting of n nodes corresponding to the agents and another part with n nodes corresponding to the bundles $Y_1, \ldots, Y_n$. There exists an edge (i, j) between (the node corresponding to) agent i and (the node corresponding to) bundle Y_j if and only if $Y_j \in \mathrm{EEFX}_i^n(\mathcal{M})$.*

We abuse the notation and refer to the "nodes corresponding to agents" as "agents" and also refer to the "nodes corresponding to bundles" as "bundles". For any subsets V of nodes, $N(V)$ is the set of all neighbours of the nodes in V. For a matching M, $V(M)$ is the set of vertices of M.

8.1.1 Polynomial Local Search (PLS)

The following description of the complexity class PLS is taken from Section 7.2 in [8].

"The class PLS (Polynomial Local Search) was defined by [50] to capture the complexity of finding local optima of optimization problems. Here, a generic instance $\mathcal{I}$ of an optimization problem has a corresponding finite set of solutions $S(\mathcal{I})$ and a potential $c(s)$ associated with each solution $s \in S(\mathcal{I})$. The objective is to find a solution that maximizes (or minimizes) this potential. In the local search version of the problem, each solution $s \in S(\mathcal{I})$ additionally has a well-defined neighborhood $N(s) \in 2^{S(\mathcal{I})}$ and the objective is to find a local optimum, i.e., a solution $s \in S(\mathcal{I})$ such that no solution in its neighborhood $N(s)$ has a higher potential.

[1] PROP1 is another relaxation of proportionality which requires that all agents receive their proportional share after adding *some* good that is not yet allocated to them.

Definition 8.3 (PLS). *Consider an optimization problem $\mathcal{X}$, and for all input instances $\mathcal{I}$ of $\mathcal{X}$ let $S(\mathcal{I})$ denote the finite set of feasible solutions for this instance, $N(s)$ be the neighborhood of a solution $s \in S(\mathcal{I})$, and $c(s)$ be the potential of solution s. The desired output is a local optimum with respect to the potential function.*

Specifically, $\mathcal{X}$ is a polynomial local search problem (i.e., $\mathcal{X} \in$ PLS) if all solutions are bounded in the size of the input $\mathcal{I}$ and there exists polynomial-time algorithms $\mathcal{A}_1$, $\mathcal{A}_2$, and $\mathcal{A}_3$ such that:

(1) $\mathcal{A}_1$ tests whether the input $\mathcal{I}$ is a legitimate instance of $\mathcal{X}$ and if yes, outputs a solution $s_{initial} \in S(\mathcal{I})$.

(2) $\mathcal{A}_2$ takes as input instance $\mathcal{I}$ and candidate solution s, tests if $s \in S(\mathcal{I})$ and if yes, computes $c(s)$.

(3) $\mathcal{A}_3$ takes as input instance $\mathcal{I}$ and candidate solution s, tests if s is a local optimum and if not, outputs $s' \in N(s)$ such that $c(s') > c(s)$ (the inequality is reversed for the minimization version).

Each PLS problem comes with an associated local search algorithm that is implicitly described by the three algorithms mentioned above. The first algorithm is used to find an initial solution to the problem and the third algorithm is iteratively used to find a potential-improving neighbor until a local optimum is reached."

8.2 Existence of Epistemic EFX Allocations

In this section, we prove the existence of EEFX allocations for any fair division instance with n agents having monotone valuations. We start by proving an important structural property (in Lemma 8.4) that enables us to reduce an instance to one with lower number of agents.

Lemma 8.4. *For a fair division instance, consider a partial allocation $(X_{k+1}, X_{k+2}, \ldots, X_n)$ to agents in the set $[n] \setminus [k]$ such that for all agents $j \in [n] \setminus [k]$ and all $i \in [k]$, we have $X_j \in \text{EEFX}_j^n(\mathcal{M})$, $X_j \notin \text{EEFX}_i^n(\mathcal{M})$. If $(X_1, \ldots, X_k)$ is an* EEFX *allocation of $\mathcal{M} \setminus \bigcup_{\ell \in [k]} X_\ell$ for agents in $[k]$, then $(X_1, X_2, \ldots, X_n)$ is an* EEFX *allocation for agents in $[n]$.*

Proof. Since for all agents $j \in [n] \setminus [k]$ we have $X_j \in \text{EEFX}_j^n(\mathcal{M})$, it suffices to prove $X_i \in \text{EEFX}_i^n(\mathcal{M})$ for all $i \in [k]$. For all $i \in [k]$, there exists a k-certificate of X_i for i under $\mathcal{M} \setminus \bigcup_{\ell \in [k]} X_\ell$, we call it $C = \{C_1, \ldots, C_{k-1}\}$. Without loss of generality, assume $v_i(X_n) \geq \ldots \geq v_i(X_{k+1})$. If $v_i(X_n) \geq v_i(X_i)$, then $(C_1, \ldots, C_{k-1}, X_i, X_{k+1}, \ldots, X_{n-1})$ is an n-certificate of X_n for i under $\mathcal{M}$. This is a contradiction with $X_n \notin \text{EEFX}_i^n(\mathcal{M})$. Therefore, $v_i(X_i) \geq v_i(X_n) \geq v_i(X_\ell)$ for all $\ell \in [n] \setminus [k]$. Hence $(C_1, \ldots, C_{k-1}, X_{k+1}, \ldots, X_n)$ is an n-certificate of X_i for i under $\mathcal{M}$ and thus, $X_i \in \text{EEFX}_i^n(\mathcal{M})$. □

We will now give a high-level overview of our constructive proof for establishing the existence of EEFX allocation among arbitrary number of agents with monotone valuations using ALG (see Algorithm 11). For a fair division instance $\mathcal{I} = (\mathcal{N}, \mathcal{M}, \mathcal{V})$, our algorithm ALG, starts by considering an EFX allocation $(X_1, \ldots, X_n)$ of $\mathcal{M}$ among n agents with

Input: A fair division instance $\mathcal{I} = (\mathcal{N}, \mathcal{M}, \mathcal{V})$ where agent $i \in \mathcal{N} = [n]$ has monotone valuation v_i over the set of items $\mathcal{M}$.
Output: An allocation $X = (X_1, X_2, \ldots, X_n)$.

if $\mathcal{N} = \emptyset$ **then**
    return $\emptyset$;
$n \leftarrow |\mathcal{N}|$
$(X_1, \ldots, X_n) \leftarrow$ an EFX allocation of $\mathcal{M}$ among n agents with valuation v_n;
$G \leftarrow$ EEFX-graph of $\{X_1, \ldots, X_n\}$;
Let $M = \{(k+1, X_{k+1}), \ldots, (n, X_n)\}$ be a matching of size at least 1 such that $N(\{X_{k+1}, \ldots, X_n\}) = \{k+1, \ldots, n\}$;
$\mathcal{N}' \leftarrow [k]$;
$\mathcal{M}' \leftarrow \mathcal{M} \setminus \bigcup_{\ell \in [n] \setminus [k]} X_\ell$;
$\mathcal{V}' \leftarrow (V_1, \ldots, V_k)$;
$(X_1, \ldots, X_k) \leftarrow \text{EEFX}(\mathcal{N}', \mathcal{M}', \mathcal{V}')$;
return $(X_1, X_2, \ldots, X_n)$;

Algorithm 11: $\mathsf{ALG} = \text{EEFX}(\mathcal{I})$

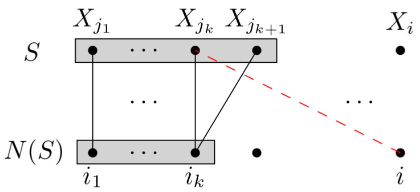


Figure 8.1: If $G(X)$ does not admit a perfect matching, then there exists a minimal subset $S = \{X_{j_1}, \ldots, X_{j_{k+1}}\}$ of bundles such that $|N(S)| < k+1$. Then, for all agent $i \in N(S)$ and all $\ell \in [k+1]$, no edge between X_ℓ and i exists. In other words, no such red dashed edges can exist.

valuation v_n. We know such an allocation exists by the work of [59]. Next, we construct the EEFX-graph G between the bundles $X_1, \ldots, X_n$ and the agents. Lemma 8.5 proves that there will always exist a non-trivial matching $M = \{(k+1, X_{k+1}), \ldots, (n, X_n)\}$[2] such that $N(\{X_{k+1}, \ldots, X_n\}) = \{k+1, \ldots, n\}$. That is, for every $j \in [n] \setminus [k]$, bundle $X_j \in \text{EEFX}_j^n(\mathcal{M})$.

Next, ALG reduces the instance by removing the agents $\{k+1, k+2, \ldots, n\}$ from $\mathcal{N}$ with their bundles $X_{k+1}, X_{k+2}, \ldots, X_n$ safely. Note that, no agent $i \in [k]$ has any edge in G to any bundle X_j for $j \in [n] \setminus [k]$. Finally, this also implies that finding an EEFX allocation $(X_1, X_2, \ldots, X_k)$ in the reduced instance and combining it with $(X_{k+1}, X_{k+2}, \ldots, X_n)$ leads to an overall EEFX allocation in the original instance. That is, our technique enables us to reduce our instance, find an EEFX allocation in the reduced instance, and combine it in such a way that we produce an EEFX allocation for the original instance.

We begin by proving Lemma 8.5.

[2]Without loss of generality, we can rename the bundles and agents in the matching M

Lemma 8.5. *For any fair division instance, consider an agent $i \in \mathcal{N}$, let $(X_1, \ldots, X_n)$ be an* EFX *allocation for an instance consisting of n agents with identical valuations v_i. Let G be the* EEFX*-graph with n agents and n bundles $X_1, \ldots, X_n$. Then there always exists a matching $M = \{(i_1, X_{j_1}), \ldots, (i_k, X_{j_k})\}$ of size at least 1, such that $N(\{X_{j_1}, \ldots, X_{j_k}\}) = \{i_1, \ldots, i_k\}$.*

Proof. To begin with, if G has a perfect matching $M = \{(i_1, X_{j_1}), \ldots, (i_n, X_{j_n})\}$, then the lemma trivially holds true.

Therefore, let us assume that no perfect matching exists in G. This implies that the Hall's condition is not satisfied, i.e., there exists a subset $S = \{X_{j_1}, \ldots, X_{j_{k+1}}\}$ of bundles such that $|N(\{X_{j_1}, \ldots, X_{j_{k+1}}\})| < k + 1$. See Figure 8.1 for a better intuition. We assume that the subset $S = \{X_{j_1}, \ldots, X_{j_{k+1}}\}$ is minimal. That is, for all $S' \subsetneq S$, we have $N(S') \geq |S'|$. Now consider $T = \{X_{j_1}, \ldots, X_{j_k}\} \subsetneq S$. By minimality of S, we know that Hall's condition holds for T, i.e., there exists a perfect matching, say $M = \{(i_1, X_{j_1}), \ldots, (i_k, X_{j_k})\}$ between the nodes in T and $N(T)$. Since $|N(S)| < k + 1$ and $\{i_1, \ldots, i_k\} \subseteq N(T) \subseteq N(S)$, it follows that $N(S) = N(T) = \{i_1, \ldots, i_k\}$.

Note that since $(X_1, \ldots, X_n)$ is an EFX allocation for an instance with identical valuations v_i, we know that $i \in N(S)$, thus $k \geq 1$. Hence, $M = \{(i_1, X_{j_1}), \ldots, (i_k, X_{j_k})\}$ is a matching of size $k \geq 1$, such that $N(\{X_{j_1}, \ldots, X_{j_k}\}) = \{i_1, \ldots, i_k\}$. The stated claim stands proven. □

Theorem 8.6 ([59])**.** *When agents have identical monotone valuations, there always exists an* EFX *allocation.*

We are now ready to discuss our main result that constructively establishes the existence of EEFX allocation among arbitrary number of agents with monotone valuations using ALG.

Theorem 8.7. EEFX *allocations exist for any fair division instance with monotone valuations. In particular,* ALG *returns an* EEFX *allocation.*

Proof. We begin by proving that Algorithm 11 terminates. By Theorem 8.6 and Lemma 8.5, a matching $M = \{(i_1, X_{i_1}), \ldots, (i_t, X_{i_t})\}$ of size at least 1 exists such that $N(\{X_{i_1}, \ldots, X_{i_t}\}) = \{i_1, \ldots, i_t\}$. Note that we can rename the bundles and the agents and without loss of generality assume that the considered matching is $M = \{(k + 1, X_{k+1}), \ldots, (n, X_n)\}$. Therefore, after removing $\{k + 1, \ldots, n\}$ from $\mathcal{N}$, the size of $\mathcal{N}$ decreases. Hence, the depth of the recursion is bounded by n (the initial number of agents).

We prove the correctness of ALG by using induction on the number of the agents. If $\mathcal{N} = \emptyset$, then $\emptyset$ is an EEFX allocation. We assume that ALG returns an EEFX allocation for any fair division instance with $n' < n$ agents with monotone valuations. Consider the matching M described in ALG. We will show the output allocation of ALG for n agents is EEFX as well. For all $j \in [n] \setminus [k]$ and all $i \in [k]$, the matching M ensures that we have $X_j \in \text{EEFX}_j^n(\mathcal{M})$, and $X_j \notin \text{EEFX}_i^n(\mathcal{M})$ (see Figure 8.1). By induction hypothesis, $(X_1, \ldots, X_k)$ is an EEFX allocation of $\mathcal{M} \setminus \bigcup_{\ell \in [k]} X_\ell$ for agents in $[k]$. Thus, by Lemma 8.4, $(X_1, X_2, \ldots, X_n)$ is an EEFX allocation for agents in $[n]$. □

8.3 Hardness Results

In this section, we complement our existential result of EEFX allocations for monotone valuations by proving computational and information-theoretic lower bounds for finding an EEFX allocation. When agents have submodular valuation functions, the way to compute the value $v(S)$ for a subset S of the items is through making value queries. [59] proved that exponentially many value queries are required to compute an EFX allocation *even* for two agents with identical submodular valuations. Formally, they proved the following information-theoretic lower bounds.

Theorem 8.8 ([59])**.** *The query complexity of finding an* EFX *allocation with* $|\mathcal{M}| = 2k+1$ *many items is* $\Omega(\frac{1}{k}\binom{2k+1}{k})$*, even for two agents with identical submodular valuations.*

Moreover, [45] proved the following computational hardness for EFX allocations.

Theorem 8.9 ([45])**.** *The problem of computing an* EFX *allocation for two agents with identical submodular valuations is* PLS*-complete.*

Now we define the computational problems corresponding to finding EFX and EEFX allocations.

Definition 8.10. *(*ID-EFX*) Given a fair division instance* $\mathcal{I} = ([2], \mathcal{M}, (v, v))$ *with two agents having identical submodular valuations* v*, find an* EFX *allocation.*

Definition 8.11. *(*ID-EEFX*) Given a fair division instance* $\mathcal{I} = ([n], \mathcal{M}, (v, \dots, v))$ *with* n *agents having identical submodular valuations* v*, find an* EEFX *allocation.*

We reduce the problem of finding an EFX allocation for two agents with identical submodular valuations (ID-EFX) to finding an EEFX allocation for an arbitrary number of agents with identical submodular valuations (ID-EEFX), thereby establishing similar hardness results for the latter.

Our Reduction: Consider an arbitrary instance $\mathcal{I} = ([2], \mathcal{M}, (v, v))$ of ID-EFX with two agents having identical submodular valuations v. Let $\mathcal{I}' = ([n], \mathcal{M}', (v', \dots, v')\})$ be an instance of ID-EEFX with n agents having identical valuations v' over the set of items $\mathcal{M}' = \mathcal{M} \cup \{h_1, \dots, h_{n-2}\}$. We define the valuation v' as follows.

- For all $S \subseteq \mathcal{M}$, $v'(S) = v(S)$.
- For all $j \in [n-2]$, $v'(h_j) = 2v(\mathcal{M}) + 1$.
- For all $j \in [n-2]$ and $S \subseteq \mathcal{M}' \setminus \{h_j\}$, $v'(S \cup \{h_j\}) = v(S) + v(h_j)$.

We call items $h_1, \dots, h_{n-2}$ heavy items. Note that we can compute $\mathcal{I}'$ from $\mathcal{I}$ in polynomial time.

Lemma 8.12. *If* v *is a submodular function, then* v' *is a submodular function as well.*

Proof. We need to prove that for all $S, T \subseteq \mathcal{M}$, $v'(S) + v'(T) \geq v'(S \cup T) + v'(S \cap T)$. Let H_S and H_T be the set of all heavy items in S and T respectively. We have

$$
\begin{aligned}
& v'(S) + v'(T) \\
& = v'(S \setminus H_S) + v'(H_S) + v'(T \setminus H_T) + v'(H_T) \\
& = (v(S \setminus H_S) + v(T \setminus H_T)) + v'(H_S) + v'(H_T)) \\
& \geq v((S \setminus H_S) \cup (T \setminus H_T)) + v((S \setminus H_S) \cap (T \setminus H_T)) \\
& \quad + v'(H_S) + v'(H_T) && \text{(submodularity of } v) \\
& = v'((S \cup T) \setminus (H_S \cup H_T)) + v'((S \cap T) \setminus (H_S \cap H_T)) \\
& \quad + v'(H_S) + v'(H_T) \\
& = v'((S \cup T) \setminus (H_S \cup H_T)) + v'((S \cap T) \setminus (H_S \cap H_T)) \\
& \quad + v'(H_S \cup H_T) + v'(H_S \cap H_T) && \text{(additivity of } v' \text{ on heavy items)} \\
& = v'(S \cup T) + v'(S \cap T).
\end{aligned}
$$

□

Lemma 8.13. *Given any* EEFX *allocation A in $\mathcal{I}'$, we can create an* EFX *allocation in $\mathcal{I}$ in polynomial time, where $\mathcal{I}$ and $\mathcal{I}'$ are as defined above.*

Proof. Let us assume that $A = (A_1, \ldots, A_n)$ is an EEFX allocation in instance $\mathcal{I}'$. To begin with, note that there are $n-2$ heavy items in $\mathcal{I}'$, and hence, by pigeonhole principle, there exists at least two agents, say $i, j \in \mathcal{N}'$ such that they receive no heavy item under A. Without loss of generality, let us assume that $i = 1$ and $j = 2$, and hence we have $A_1, A_2 \subseteq \mathcal{M}$. This implies that we have

$$
\begin{aligned}
v'(A_1) = v(A_1),\ & v'(A_2) = v(A_2), \\
\text{and, } v(A_1), v(A_2) & < 2v(\mathcal{M}) + 1
\end{aligned} \tag{8.1}
$$

Without loss of generality, let us assume $v(A_2) \geq v(A_1)$.

We will prove $(A_1, \mathcal{M} \setminus A_1)$ forms an EFX allocation in $\mathcal{I}$. Note that valuations v and v' coincide for the bundles A_1 and $\mathcal{M} \setminus A_1$. Since A is EEFX in $\mathcal{I}'$, let us denote the n-certificate for agent 1 with respect to A_1 by $C = (C_2, C_3, \ldots, C_n)$. First, we prove that no bundle C_k with a heavy item can have any other item as well. Assume otherwise. Let $\{g, h_j\} \subseteq C_k$ for some $k \in \{2, \ldots, n\}$ and some $j \in [n-2]$ and $g \neq h_j$. Then, we have

$$v'(C_k \setminus g) \geq v'(h_j) = 2v(\mathcal{M}) + 1 > v'(A_1)$$

where, the last inequality uses equation (8.1). This implies that agent 1 strongly envies bundle C_k which is a contradiction to our assumption that C forms an n-certificate for bundle A_1 in instance I'. Therefore, the $n-1$ bundles in the n-certificate must look like $\{C_2, \ldots, C_n\} = \{\{h_1\}, \ldots, \{h_{n-2}\}, \mathcal{M} \setminus A_1\}$. First, note that, agent 1 with bundle A_1 must not strongly envy bundle $\mathcal{M} \setminus A_1$ since C is an n-certificate. And since, we already have $v(A_2) \geq v(A_1)$ and $A_2 \subseteq \mathcal{M} \setminus A_1$, the allocation $(A_1, \mathcal{M} \setminus A_1)$ forms an EFX allocation in $\mathcal{I}$. □

Theorem 8.14. *The query complexity of the* EEFX *allocation problem with $|\mathcal{M}| = 2k + n - 1$ many items is $\Omega(\frac{1}{k}\binom{2k+1}{k})$, for arbitrary number of agents n with identical submodular valuations.*

Proof. Consider any arbitrary instance $\mathcal{I} = ([2], \mathcal{M}, (v, v))$ with two agents having identical submodular valuations v and $|\mathcal{M}| = 2k + 1$ items. Create the instance $\mathcal{I}'$ as described above. Using Lemma 8.12, $\mathcal{I}'$ consists of n agents with identical submodular valuations. By Lemma 8.13, given any EEFX allocation A, we can obtain an EFX allocation for $\mathcal{I}$ in polynomial time. Finally, using Theorem 8.8, we know that the query complexity of finding an EFX allocation in $\mathcal{I}$ is $\Omega(\frac{1}{k}\binom{2k+1}{k})$. Hence, the query complexity of EEFX for n agents with identical submodular valuations admits the same lower bound. This establishes the stated claim. □

Finally, our next result follows using Lemma 8.13 and Theorem 8.9.

Theorem 8.15. *The problem of computing an* EEFX *allocation for arbitrary number of agents with identical submodular valuations is* PLS*-hard.*

Since our reduction work even for three agents, Theorems 8.14 and 8.15 hold true for the problem of computing EEFX allocations even for three agents with identical submodular valuations. Note that the set of EFX and EEFX allocations coincide for the case of two agents and hence it inherits the same computational hardness guarantees as that of EFX here.

PART III

Simultaneous Fairness Guarantees

CHAPTER 9

Achieving MMS and EFX/EF1 Guarantees Simultaneously

We study fair division instances with agents having additive valuations over a set of indivisible items. The aim of this work is to push our understanding of the compatibility between two different classes of fairness notions: EFX/EF1 with MMS guarantees. Our main contribution is developing (simple) algorithms for achieving EFX/EF1 and MMS guarantees simultaneously.

Main Theorem: For any fair division instance, we show that there exists

(1) a partial allocation that is both 2/3-MMS and EFX [see Theorem 9.10 and Algorithm 13].

(2) a complete allocation that is both 2/3-MMS and EF1 [see Theorem 9.16 and Algorithm 15].

We note that the latter is an immediate corollary of the former using known techniques [55]. For completeness, in Section 9.4 we provide the full proof.

If we relax 2/3-MMS to $(2/3 - \varepsilon)$-MMS for any arbitrary constant $\varepsilon > 0$, then the above allocations can be computed in pseudo-polynomial time. If in addition to that, we relax EFX/EF1 to $(1 - \delta)$-EFX/$(1 - \delta)$-EF1, then the allocations can be computed in polynomial time.

We use Algorithm 12 developed by [4] to compute 2/3-MMS allocations as a starting point to have share-based guarantee. Here, as soon as an agent receives a bundle, she is taken out of consideration. This feature of the algorithm is incompatible with achieving any envy-based guarantees.[1] We overcome this barrier and develop a novel algorithm (Algorithm 13) that removes the *myopic* nature of Algorithm 12 and also looks into the future and modifies the already-allocated bundles if needed. Interestingly enough, the share-based guarantee that we maintain for a subset of agents (whose size keep growing) throughout the execution of Algorithm 13 helps us to prove envy-based guarantees as well.

Our first result improves the guarantees shown by [30] where they develop a pseudo-polynomial time algorithm to compute a partial allocation that is both 1/2-MMS and EFX. Also, [7] develop an efficient algorithm to compute a complete allocation that is simultaneously 0.553-MMS and 0.618-EFX; note that, this is incomparable to the guarantees that we develop in this work. On the other hand, the best known approximation factors, prior to our work, for simultaneous guarantees on MMS and EF1 was by [7] where they efficiently find allocations that are 4/7-MMS and EF1.

[1] We note that this feature is common to many other algorithms achieving share-based guarantees in the fair division literature. See "valid reductions" in Chapter 3.

9.1 Notations and Tools

Proposition 9.1 ([71]). *Given any fair division instance with additive valuations, there exists a PTAS to compute an* MMS*-partition of any agent $i \in \mathcal{N}$, and hence her* MMS_i *value as well.*

We define two graphs inspired by share-based and envy-based fairness notions, that will prove useful in our algorithms.

Definition 9.2 (Threshold-Graph). *Given a partition $Y = (Y_1, \ldots, Y_n)$ of $\mathcal{M}$ into n bundles and given a vector $t = (t_1, \ldots, t_n) \in \mathbb{R}^n_{\geq 0}$, we define the* threshold-graph *as an undirected bipartite graph $T_{\langle Y,t \rangle} = (V, E)$, where V has one part consisting of n nodes corresponding to the agents and another part with n nodes corresponding to the bundles $Y_1, \ldots, Y_n$. There exists an edge (i, j) between (the node corresponding to) agent i and (the node corresponding to) bundle Y_j if and only if $v_i(Y_j) \geq t_i$. For all $i \in [n]$, we call t_i, the threshold share value of agent i.*

For a subset S of the nodes, we write $N(S)$ to denote the set of neighbours of the nodes in S in the threshold graph.

Recall from Definition 2.17 that G_X is the envy-graph of allocation X.

In this chapter, without loss of generality, by scaling the value of each item for each agent i by a factor of $1/\text{MMS}_i$, we assume that the MMS values of all the agents are 1. Note that the scaling the valuations does not change the EFX, EF1, and α-MMS properties of an allocation.

9.2 Relations Between EFX/EF1 and MMS

In this section, we briefly discuss the guarantees EFX/EF1 allocations can provide for MMS and vice versa. [5] gave a comprehensive comparison between these notions of fairness. Here we mention a few. In particular, they proved that while a complete EFX allocation implies 4/7-MMS guarantee, there exists complete EFX allocations which are as bad as 0.5914-MMS. Nevertheless, these guarantees become relevant only when a complete EFX allocation exists which, in itself, is a big open problem.

Proposition 9.3 ([5]). *For arbitrary $n \geq 1$, any* EFX *allocation is also a* 4/7-MMS *allocation. On the other hand, an* EFX *allocation is not necessarily an α-*MMS *allocation for $\alpha > 0.5914$ and large enough n.*

Proposition 9.4 ([5]). *An* EF1 *allocation is not necessarily an α-*MMS *allocation for any $\alpha > 1/n$.*

Proof. Consider the instance $\mathcal{I}$ with n agents with identical valuation v over $2n-1$ items. Assume $v(g_i) = 1$ for all $i \in [n-1]$ and $v(g_i) = 1/n$ for all $i \in [2n-1] \setminus [n-1]$. Consider the following partition of $\mathcal{M}$ into n bundles. For $i \in [n-1] : A_i = \{g_i\}$ and $A_n = \{g_n, \ldots, g_{2n-1}\}$. $v(A_i) = 1$ for all $i \in [n]$ and thus the MMS value of all agents is 1. Now consider the following allocation. $X_i = \{g_i, g_{n+i-1}\}$ for $i \in [n-1]$ and $X_n = \{g_{2n-1}\}$. For all $i \in [n-1]$, $v(X_i) = 1 + 1/n$ and $v(X_n) = 1/n$. The allocation X is EF1 since $v(X_n) \geq v(X_i \setminus \{g_i\})$ for all $i \in [n-1]$. However, X is $1/n$-MMS. ☐

Proposition 9.5. *For $n \geq 3$ and any $\alpha > 0$, an α-MMS allocation is not necessarily β-EF1 for any $\beta > 0$.*

Proof. Consider the instance $\mathcal{I}$ with $n \geq 3$ agents with identical valuation v over 2 items with $v(g_1) = v(g_2) = 1$. Clearly the MMS value of all the agents is 0 and thus all allocations are α-MMS for any $\alpha > 0$. Now consider the allocation X with $X_n = \{g_1, g_2\}$ and $X_i = \emptyset$ for all $i \in [n-1]$. For all $i \in [n-1]$ and $\beta > 0$, $v(X_i) < \beta v(X_n \setminus \{g_1\})$. Thus, X is not β-EF1. □

Therefore, we can conclude that, by guaranteeing one of approximate MMS or approximate EFX/EF1, one cannot obtain a good guarantee for the other notion of fairness for free.

9.3 $\frac{2}{3}$-MMS Together with EFX

First we describe and analyze the algorithm developed by Procaccia et al. [53, 62] and Amanatidis et al. [4] to compute 2/3-MMS allocations for fair division instances with additive valuations. We rewrite it and analyze it in our own words (in Algorithm 12) since we use it to develop our main algorithm (Algorithm 13) to compute allocations that are both 2/3-MMS and EFX.

Surprisingly, Algorithm 12 does *not* rely on the two most commonly used tools for computing approximate MMS allocations, namely *ordered instances* and *valid reductions* (see Section 2.1).

Unfortunately, these tools cannot be used when dealing with envy-based notions of fairness. And hence, most of the previous works that achieve approximate MMS guarantees do not obtain any envy-based criteria results. On the other hand, most of the previous work that achieve simultaneous guarantees for MMS and EFX/EF1 are obtained by manipulating algorithms that provide EFX/EF1 guarantees so that some approximation for MMS can also be achieved [7, 30]. However, so far, the envy-based algorithmic techniques have not been strong enough to also attain 2/3-MMS guarantee.
Overview of Algorithm 12: Algorithm 12 successively allocates a bundle of items to some selected agents in each step and removes them from consideration. In particular, in each round of Algorithm 12 with n' remaining agents, we ask a remaining agent i to divide the remaining items into n' bundles $X_1, \dots, X_{n'}$, each of value at least $2/3 \cdot \text{MMS}_i^n(\mathcal{M})$ to her. We prove, in Lemma 9.7, that the above is always possible at every step of the algorithm. Without loss of generality assume $[n']$ is the set of remaining agents. Then, we consider the threshold graph $T_{\langle X,t \rangle}$ with $X = (X_1, \dots, X_{n'})$ and $t = 2/3(\text{MMS}_1^n(\mathcal{M}), \dots, \text{MMS}_{n'}^n(\mathcal{M}))$ and find a matching between the bundles and the agents such that (i) every matched agent j has a value of at least $2/3 \cdot \text{MMS}_j^n(\mathcal{M})$ for the bundle matched to her and (ii) every unmatched agent j values any of the matched bundles at less than $2/3 \cdot \text{MMS}_j^n(\mathcal{M})$. We allocate according to this matching, and remove the matched agents with their matched bundles. As long as there is any remaining agent, we repeat the above process. See Algorithm 12 for the pseudo-code of this algorithm. A similar technique is also used in [1, 48, 51, 64] and Chapter 8.

Theorem 9.6 ([4])**.** *For fair division instances with additive valuations, Algorithm 12 returns a $\frac{2}{3}$-MMS allocation.*

Input: A fair division instance $\mathcal{I} = (\mathcal{N}, \mathcal{M}, \mathcal{V})$ with additive valuations.
Output: An allocation X.

Let $\text{MMS}_i = \text{MMS}_i^n(\mathcal{M})$ for all $i \in [n]$
while $\mathcal{N} \neq \emptyset$ **do**
  $n \leftarrow |\mathcal{N}|$
  Let $i \in \mathcal{N}$
  Let $(X_1, \ldots, X_n)$ be a partition of $\mathcal{M}$ such that $v_i(X_j) \geq \frac{2}{3}\text{MMS}_i$
  Let $T_{\langle X,t \rangle}$ be the threshold-graph with $X = (X_1, \ldots, X_n)$ and $t = \frac{2}{3}(\text{MMS}_1, \ldots, \text{MMS}_n)$ for agents in $[n]$
  Let $M = \{(k+1, X_{k+1}), \ldots, (n, X_n)\}$ be a matching of size at least 1 such that $N(\{X_{k+1}, \ldots, X_n\}) = \{k+1, \ldots, n\}$;
  $\mathcal{N} \leftarrow [k]$;
  $\mathcal{M} \leftarrow \mathcal{M} \setminus \bigcup_{\ell \in [n] \setminus [k]} X_\ell$;
return $(X_1, X_2, \ldots, X_n)$;

Algorithm 12: 2/3-`approxMMS`$(\mathcal{I})$

First we prove the following lemmas.

Lemma 9.7. *Fix an agent $i \in \mathcal{N}$ and some $k < n$. Consider k disjoint bundles $A_1, \ldots, A_k \subseteq \mathcal{M}$ such that for all $j \in [k]$, we have $v_i(A_j) < \frac{2}{3} \cdot \text{MMS}_i^n(\mathcal{M})$. Then, there exists a partition $(B_1, \ldots, B_{n-k})$ of $\mathcal{M} \setminus \cup_{j \in [k]} A_j$ into $n-k$ bundles such that $v_i(B_j) \geq \frac{2}{3} \cdot \text{MMS}_i^n(\mathcal{M})$ for all $j \in [n-k]$.*

Proof idea. We first give an overview of our idea before formally proving Lemma 9.7. Let us begin by fixing an MMS-partition of agent i in the given instance. Then, given the k bundles $A_1, \ldots, A_k \subseteq \mathcal{M}$ with the property as stated in Lemma 9.7, we categorize the bundles of this MMS-partition depending on how much value these bundles have after the removal of the items in $A_1 \cup \ldots \cup A_k$. All the bundles with remaining value of at least 2/3 are in the set C^0 and all the bundles with remaining value at least 1/3 and at most 2/3 are in the set C^1. By pairing the bundles in C^1 and merging the items in this pair, we manage to create $\lfloor \frac{|C^1|}{2} \rfloor$-many bundles of value at least 2/3. Moreover, all the bundles in C^0 are of value at least 2/3. Therefore, it is sufficient to prove $|C^0| + \lfloor \frac{|C^1|}{2} \rfloor \geq n-k$. We show it by upper-bounding the value of the removed items for agent i.

Proof. For a given fair division instance $\mathcal{I}$ and an agent $i \in \mathcal{N}$, let $(C_1, \ldots, C_n)$ be an MMS-partition of i. Let $A = A_1 \cup \ldots \cup A_k$ and write $D_j = C_j \cap A$ and $C'_j = C_j \setminus A$ for all $j \in [n]$. We define C^0, C^1, and C^2 as follows, depending on the amount of value removed from each C_j after the removal of the items in A.

- $C^0 = \{j \in [n] \mid v_i(D_j) \leq \frac{1}{3}\}$.
- $C^1 = \{j \in [n] \mid \frac{1}{3} < v_i(D_j) \leq \frac{2}{3}\}$.
- $C^2 = \{j \in [n] \mid \frac{2}{3} < v_i(D_j)\}$.

Let $n_0 = |C^0|$, $n_1 = |C^1|$, and $n_2 = |C^2|$. Note that $n = n_0 + n_1 + n_2$. Without loss of generality, we assume that $C^0 = \{1, \ldots, n_0\}$, $C^1 = \{n_0 + 1, \ldots, n_0 + n_1\}$, and

$C^2 = \{n_0 + n_1 + 1, \ldots, n\}$. We aim to create $n-k$ many bundles $B_1, \ldots, B_{n-k} \subset \mathcal{M} \setminus A$, each of value at least 2/3 to agent i. To begin with, we set $B_j = C'_j$ for all $j \in [n_0]$. Next, we pair the bundles with indices in C^1 and unite the sets in each pair. Formally, for all $j \in [\lfloor \frac{n_1}{2} \rfloor]$, we set $B_{n_0+j} = C'_{n_0+2j-1} \cup C'_{n_0+2j}$.

First, note that, for all $j \in C^0$, $v_i(B_j) = v_i(C'_j) \geq \frac{2}{3}$. Also, since for all $j \in C^1$, $v_i(C'_j) \geq \frac{1}{3}$, for all $\ell \in [n_0 + \lfloor \frac{n_1}{2} \rfloor]$, $v_i(B_j) \geq \frac{2}{3}$. Therefore, to complete our proof, it suffices to establish that $n_0 + \lfloor \frac{n_1}{2} \rfloor \geq n - k$. We have,

$$\begin{aligned}
\frac{2}{3} \cdot k &> \sum_{j \in [k]} v_i(A_j) && (v_i(A_j) < 2/3 \text{ for all } j \in [k]) \\
&= \sum_{j \in [n]} v_i(C_j \cap A) = \sum_{j \in [n]} v_i(D_j) \\
&= \sum_{j \in [n_0]} v_i(D_j) + \sum_{j \in [n_1+n_0] \setminus [n_0]} v_i(D_j) + \sum_{j \in [n_2+n_1+n_0] \setminus [n_1+n_0]} v_i(D_j) \\
&\geq \frac{1}{3} n_1 + \frac{2}{3} n_2.
\end{aligned}$$

That is, $k > \frac{n_1}{2} + n_2$, or equivalently, $n - k < n - (\frac{n_1}{2} + n_2) = n_0 + \frac{n_1}{2}$. Therefore, we have $n_0 + \lfloor \frac{n_1}{2} \rfloor \geq n - k$, as desired. □

Given a set of nodes S in a threshold graph T, $N(S)$ is the set of all neighbors of the nodes in S.

Lemma 9.8. *For a given partition $X = (X_1, \ldots, X_n)$ of $\mathcal{M}$ and a threshold vector $t = (t_1, \ldots, t_n)$, assume for all $j \in [n]$, there exists an agent i such that $v_i(X_j) \geq t_i$. Then $T_{\langle X,t \rangle}$ has a non-empty matching $M = \{(i_1, X_{j_1}), \ldots, (i_k, X_{j_k})\}$ such that $N(\{X_{j_1}, \ldots, X_{j_k}\}) = \{i_1, \ldots, i_k\}$. Moreover, M can be computed in polynomial time.*

Proof. First we compute a maximum matching M^* in $T_{\langle X,t \rangle}$ (which can be done in polynomial time). If M^* is a perfect matching between $[n]$ and $(X_1, \ldots, X_n)$, then clearly the lemma holds. Otherwise, there must exists a Hall's violator set $S \subseteq \{X_1, \ldots, X_n\}$ with $|N(S)| < |S|$. A minimal such set S can be computed in polynomial time [40]. Note that for all $X_j \in S$, there exists an agent i such that $v_i(X_j) \geq t_i$ and hence, $|S| \geq 2$. By minimality of S, the Hall's condition holds for any proper subset of S. Let $T \subset S$ and $|T| = |S| - 1$. We have $|S| - 1 = |T| \leq |N(T)| \leq |N(S)| < |S|$. Hence $|N(T)| = N(S) = |S| - 1$. Since the Halls' condition holds for T, there exists a matching M covering $N(T)$. Since $N(T) \subseteq N(S)$ and $|N(T)| = |N(S)|$, we have that M is covering $N(S)$. Therefore, there exists no edge between the agents outside M and the bundles inside M. □

Proof of Theorem 9.6 Now we are ready to prove 9.6. To begin with, note that, if Algorithm 12 terminates, each agent $i \in [n]$ is matched to (and allocated) say, bundle X_i in some threshold-graph with the threshold of agent i being $2/3 \cdot \mathrm{MMS}_i^n(\mathcal{M})$, i.e., $v_i(X_) \geq 2/3 \cdot \mathrm{MMS}_i^n(\mathcal{M})$. Thus, if the algorithm terminates, it must return a $\frac{2}{3}$-MMS allocation.

Now, in order to prove the termination of Algorithm 12, we prove that in each iteration of the while-loop, the set-size $|\mathcal{N}|$ of the remaining agents strictly decreases. Therefore, the while-loop can iterate for at most n many times and hence Algorithm 12 terminates.

Consider an arbitrary iteration of the while-loop in Algorithm 12. Let $\mathcal{N}$ and $\mathcal{M}$ be the initial set of agents and items respectively and let $\mathcal{N}' = [n']$ and $\mathcal{M}'$ be the set of remaining agents and items respectively in the beginning of this iteration of the while-loop. If $\mathcal{N}' = \emptyset$, then the algorithm obviously terminates.

Let us now consider some agent $i \in \mathcal{N}'$. First, we prove that there exists a partition $(X_1, \ldots, X_{n'})$ of $\mathcal{M}'$ such that $v_i(X_j) \geq 2/3 \cdot \text{MMS}_i^n(\mathcal{M})$ for all $j \in [n']$. We write $X_{n'+1}, \ldots, X_n$ to denote the bundles that are matched to agents $n'+1, \ldots, n$ in the previous iterations of the while-loop. Furthermore, by the choice of matching M in Step 7 of Algorithm 12, agent i did not have an edge (in the then threshold graphs) to any of these bundles $X_{n'+1}, \ldots, X_n$. That is, $v_i(X_j) < 2/3 \cdot \text{MMS}_i^n(\mathcal{M})$ for all $j \in [n] \setminus [n']$. By Lemma 9.7, there exists a partitioning $(X_1, \ldots, X_{n'})$ of $\mathcal{M}'$ such that $v_i(X_j) \geq 2/3 \cdot \text{MMS}_i^n(\mathcal{M})$ for all $j \in [n']$.

By Lemma 9.8, there exists a matching M with no edge between the agents outside M and the bundles inside M. Now without loss of generality, by renaming the agents and bundles, assume $M = \{(k+1, X_{k+1}), \ldots, (n', X_{n'}))\}$. Since we know $M \neq \emptyset$, $k < n'$ and thus the number of remaining agents decreases in the end of this iteration of the while-loop. □

Using Proposition 9.1, it is easy to see that there exists a PTAS to compute the partition $(B_1, \ldots, B_{n-k})$ in Lemma 9.7. Formally, the following lemma holds.

Lemma 9.9. *Fix an agent $i \in \mathcal{N}$ and some $k < n$. Consider k bundles $A_1, \ldots, A_k \subseteq \mathcal{M}$ such that for all $j \in [k]$, we have $v_i(A_j) < \frac{2}{3} \cdot \text{MMS}_i^n(\mathcal{M})$ for agent i. Then, a partition $(B_1, \ldots, B_{n-k})$ of the remaining items in $\mathcal{M} \setminus \cup_{j \in [k]} A_j$ into $n-k$ bundles can be computed in polynomial time such that $v_i(B_j) \geq (\frac{2}{3} - \varepsilon) \cdot \text{MMS}_i^n(\mathcal{M})$ for all $j \in [n-k]$ and all constant $\varepsilon > 0$.*

Proof. In the proof of Lemma 9.7, if an MMS-partition of i is given, computing the bundles in C^j for all $j \in \{0, 1, 2\}$ and consequently obtaining $(B_1, \ldots, B_{n-k})$ can be done in polynomial time. Fix a constant $\varepsilon > 0$ and an agent i. By Proposition 9.1, a $(1-\varepsilon/2)$-MMS partition of i can be computed in polynomial time and following the same arguments, a partition $(B_1, \ldots, B_{n-k})$ of the remaining items into $n-k$ bundles can be computed in polynomial time such that $v_i(B_j) \geq (\frac{2}{3} - \varepsilon) \cdot \text{MMS}_i^n(\mathcal{M})$. □

In the rest of this section, we modify Algorithm 12 such that the output is a (partial) allocation which is still 2/3-MMS and now becomes EFX as well. Note that, in Algorithm 12 and generally in the algorithmic technique of [4], once an agent receives a bundle X_i, X_i becomes her bundle in the final output allocation. So, once agent i receives the bundle X_i, she is out of the consideration. This guarantees that agent i will have the same utility $v_i(X_i)$ in the end of the algorithm but it does not guarantee anything about how much i values other bundles formed once she is removed from consideration. And, therefore, it cannot not guarantee EFX (or even EF1) property.

We overcome this barrier by developing Algorithm 13 in this section. Here, we again allocate a bundle of items to some selected agents in each step, but we modify them carefully in a later stage. As we will describe next, this feature of our algorithm removes the *myopic* nature of Algorithm 12 and lets us achieve envy-based fairness guarantees, while maintaining 2/3-MMS guarantees.

Overview of Algorithm 13: In each round of Algorithm 13 with $n' \leq n$ remaining agents, we ask a remaining agent i to partition the remaining goods into n' bundles $X_1, \ldots, X_{n'}$ of value at least $2/3 \cdot \text{MMS}_i^n(\mathcal{M})$. We prove, in Lemma 9.7, that it is always feasible to perform the above process at every step of the algorithm. We then shrink these bundles to guarantee that every remaining agent values each strict subset of these bundles less than 2/3 fraction of their MMS value. For simplicity, we rename the shrinked bundles again as $X_1, \ldots, X_{n'}$.

Now, let us assume that, after the process of shrinking, we still have an agent j who was allocated a bundle in previous iterations and who strongly envies one of $X_1, \ldots, X_{n'}$, say, for instance, X_j. Let us denote a^* to be a most envious agent of X_j. We allocate, to a^*, a subset of X_j which a^* envies but no agent strongly envies. In this way, we guarantee two things at each point during the algorithm, the current (partial) allocation among the agents who received a bundle so far is (a) EFX and (b) all these agents receive 2/3 fraction of their (original) MMS value. See Algorithm 13 for the pseudocode.

To the best of our knowledge, none of the previous algorithms computing an EFX allocation allocates a bundle to some of the agents and nothing to the rest in an intermediate step. It might also seem counter-intuitive to do so, since we need to guarantee that there are enough items left to satisfy the agents who have received nothing so far. We are able to make it possible in Algorithm 13, since we know that all the remaining agents (who have not yet received anything) value all the already allocated bundles less than 2/3 fraction of their MMS value. Interestingly enough, the share-based guarantee that we are maintaining helps us to prove envy-based guarantees as well.

Theorem 9.10. *For any fair division instance with additive valuations, Algorithm 13 returns a (partial) allocation that is both* EFX *and* 2/3-MMS.

Proof. We will begin by proving the correctness of Algorithm 13, and then prove that it always terminates.

Consider any arbitrary iteration of the while-loop during the execution of Algorithm 13. Let us assume there are n' remaining agents at the start of this iteration. Without loss of generality, we can rename these remaining agents as $1, 2, \ldots, n'$. This means that every agent $i \in [n] \setminus [n']$, has been assigned some bundle, say X_i. We begin by proving that $v_i(X_i) \geq 2/3 \cdot \text{MMS}_i^n(\mathcal{M})$ and that agent i does not strongly envy any agent $j \in [n] \setminus [n']$. Moreover, for all $\ell \in [n']$, $v_\ell(X_i) < 2/3 \cdot \text{MMS}_\ell^n(\mathcal{M})$.

We establish the above claim by induction. Since, initially no agent is assigned any bundle, the claim holds. Now, as the induction hypothesis, we assume that agents in $[n] \setminus [n']$ are already assigned a bundle and the (partial) allocation restricted to them is 2/3-MMS and EFX. Any change in the bundles as a result of the current while-loop can be examined by the following two cases: either the if-condition in Line 9 is satisfied, and hence, only the bundle of agent a^* changes in this iteration, or the if-condition in Line 9 is not satisfied.

- *Case 1: The if-condition in Line 9 is satisfied.* First, note that, only the bundle of agent a^* changes in this case. Let X_{a^*} and X'_{a^*} be the bundle of agent a^* before

Input: A fair division instance $\mathcal{I} = (\mathcal{N}, \mathcal{M}, \mathcal{V})$ with additive valuations.
Output: An allocation X.

1: Let $\text{MMS}_i = \text{MMS}_i^n(\mathcal{M})$ for all $i \in [n]$
2: $\mathcal{N}' \leftarrow [n]$
3: **while** $\mathcal{N}' \neq \emptyset$ **do**
4: $\quad n' \leftarrow |\mathcal{N}'|$
5: $\quad$ Let $i \in \mathcal{N}'$
6: $\quad$ Let $(X_1, \dots, X_{n'})$ be a partition of $\mathcal{M}$ such that $v_i(X_j) \geq \frac{2}{3}\text{MMS}_i$
7: $\quad$ **for** $j \in [n']$ **do**
8: $\qquad X_j \leftarrow$ minimal subset of $X'_j \subseteq X_j$ such that $\exists i' \in [n']$ with $v_{i'}(X'_j) \geq \frac{2}{3}\text{MMS}_{i'}$
9: $\qquad$ **if** $\exists a \in [n] \setminus [n']$ such that a strongly envies X_j **then**
10: $\qquad\quad$ Let $a^* \in [n] \setminus [n']$ be a most envious agent of X_j
11: $\qquad\quad$ Let $X'_j \subsetneq X_j$ be minimal such that $v_{a^*}(X'_j) > v_{a^*}(X_{a^*})$ and no agent strongly envies X'_j
12: $\qquad\quad \mathcal{M} \leftarrow \mathcal{M} \cup X_{a^*} \setminus X'_j$
13: $\qquad\quad X_{a^*} \leftarrow X'_j$
14: $\qquad\quad$ Go to Line 3
15: $\quad$ Let $T_{\langle X,t \rangle}$ be the threshold-graph with $X = (X_1, \dots, X_{n'})$ and $t = \frac{2}{3}(\text{MMS}_1, \dots, \text{MMS}_{n'})$ for agents in $[n']$
16: $\quad$ Let $M = \{(k+1, X_{k+1}), \dots, (n', X_{n'})\}$ be a matching of size at least 1 such that $N(\{X_{k+1}, \dots, X_n\}) = \{k+1, \dots, n\}$ and X_j is matched to j for all $j \in [n] \setminus [k]$;
17: $\quad \mathcal{N}' \leftarrow [k]$;
18: $\quad \mathcal{M} \leftarrow \mathcal{M} \setminus \bigcup_{\ell \in [n] \setminus [k]} X_\ell$;
return $(X_1, X_2, \dots, X_n)$;

Algorithm 13: `approxMMSandEFX`$(\mathcal{I})$

and after this iteration of the while-loop respectively. By Line 11, we know that $v_{a^*}(X'_{a^*}) > v_{a^*}(X_{a^*}) \geq \frac{2}{3}\text{MMS}_{a^*}$, and hence, the allocation restricted to $[n] \setminus [n']$ is still 2/3-MMS. Moreover, by the choice of X'_{a^*} in Line 10, no agent in $[n] \setminus [n']$ strongly envies X'_{a^*}. Since a^* did not strongly envy anyone while owning X_{a^*}, she still does not strongly envy anyone while owning X'_{a^*}. Hence, the allocation restricted to the set of agents in $[n] \setminus [n']$ is EFX and 2/3-MMS. Moreover, since X'_{a^*} is a strict subset of a minimal subset that was of value at least $2/3 \cdot \text{MMS}_\ell^n(\mathcal{M})$ to any $\ell \in [n']$, $v_\ell(X'_{a^*}) < 2/3 \cdot \text{MMS}_\ell^n(\mathcal{M})$.

- *Case 2: The if-condition in Line 9 is not satisfied.* Using Lemma 9.8, we know that the threshold-graph considered in Line 15 contains a matching M of size at least one, such that, no unmatched agent has an edge to a matched bundle.

  Now, without loss of generality, we rename the agents and bundles such that $M = \{(k+1, X_{k+1}), \dots, (n', X_{n'}))\}$. Therefore, agents in the set $[n] \setminus [k]$ hold some non-empty bundle. Note that, by induction hypothesis and by the definition of the threshold-graph, we know that for all agents $i \in [n] \setminus [k]$, we have $v_i(X_i) \geq \frac{2}{3}\text{MMS}_i$.

  Therefore, it remains to prove that the allocation restricted to agents in $[n] \setminus [k]$ is EFX as well. We split these agents into the set $[n] \setminus [n']$ and $[n'] \setminus [k]$. By induction hypothesis, we already know the allocation restricted to $[n] \setminus [n']$ is EFX. Next, since the if-condition in Line 9 is not satisfied, no agent in $[n] \setminus [n']$ strongly envies any agent in $[n'] \setminus [k]$. For all $i \in [n'] \setminus [k]$, we have $v_i(X_i) \geq 2/3 \cdot \text{MMS}_i^n(\mathcal{M})$ and $v_i(X_j) < 2/3 \cdot \text{MMS}_i^n(\mathcal{M})$ for all $j \in [n] \setminus [n']$. Hence, no agent in $[n'] \setminus [k]$ envies any agent in $[n] \setminus [n']$. Also, for all $i, j \in [n'] \setminus [k]$ and all $X'_j \subsetneq X_j$, we have $v_i(X'_j) < 2/3 \cdot \text{MMS}_i^n(\mathcal{M})$ (see Line 8). Since $v_i(X_i) \geq 2/3 \cdot \text{MMS}_i^n(\mathcal{M})$, i does not strongly envy j.

Finally, we will now prove that Algorithm 13 terminates and allocates a non-empty bundle to all agents. Let us write A to denote the set of agents who are allocated a non-empty bundle at any point during the execution of Algorithm 13. We will prove that after each iteration of the while-loop, the vector $(\sum_{i \in A} v_i(X_i), |A|)$ increases lexicographically, and hence, the algorithm must terminate. In Case 1, the utility of a^* increases while the utility of all other agents in A does not change and also $|A|$ does not change. On the other hand, in Case 2, since the matching M found in Line 16 is of size at least one, at least one more agent is added to the set A and thus $|A|$ increases. Since, all agents who were previously in A, remain in A and their utilities do not change, the claim follows. □

Note that, the vector $(\sum_{i \in A} v_i(X_i), |A|)$ can take pseudo-polynomially many values, and the only steps in Algorithm 13 that cannot be executed in polynomial time are related to computing the exact MMS values of agents and the construction of the bundles $X = (X_1, \ldots, X_{n'})$ such that $v_i(X_j) \geq (2/3)\text{MMS}_i$ in Line 6 of the while-loop. However, by Proposition 9.1 and Lemma 9.9, if we replace the MMS bound 2/3 with $2/3 - \varepsilon$ for any constant $\varepsilon > 0$, these steps can be executed in polynomial time. Therefore, we obtain the following result.

Theorem 9.11. *For fair division instances with additive valuations and any constant $\varepsilon > 0$, a (partial) allocation that is both* EFX *and* $(2/3 - \varepsilon)$-MMS *can be computed in pseudo-polynomial time.*

The only reason why the algorithm runs pseudo-polynomial time and not polynomial time, is that $\sum_{i \in A} v_i(X_i)$ in $(\sum_{i \in A} v_i(X_i), |A|)$ can take pseudo-polynomially many values. By relaxing the notion of exact EFX to $(1 - \delta)$-EFX for any constant δ, we make sure that $v_i(X_i)$ can improve $\log_{1/(1-\delta)}(v_i(\mathcal{M}))$ many times which bounds the total number of rounds polynomially.

Theorem 9.12. *For fair division instances with additive valuations and any constant $\delta > 0$ and $\varepsilon > 0$, a (partial) allocation that is both* $(1 - \delta)$-EFX *and* $(2/3 - \varepsilon)$-MMS *can be computed in polynomial time.*

9.3.1 2/3-MMS and EFX with bounded Charity

In this section, we show that we can bound the number and the value of items that go unallocated in Algorithm 13. We do so by using the algorithm `EFXwithCharity` developed by [30] which takes a partial allocation Y as input and outputs a (partial) EFX allocation X with the properties mentioned in Theorem 9.13. Recall that for a partial allocation X, $P(X)$ is the set of unallocated goods.

Theorem 9.13. *[30] Given a (partial)* EFX *allocation Y, there exists a (partial)* EFX *allocation $X = (X_1, \ldots, X_n)$, such that for all $i \in [n]$*

*(1) X is $\frac{1}{2-|P(X)|/n}$-*MMS*, and*

(2) $v_i(X_i) \geq v_i(Y_i)$, and

(3) $v_i(X_i) \geq v_i(P(X))$, and

(4) $|P(X)| < s$,

where s is the number of sources in the envy-graph of X.

Therefore, if we run `EFXwithCharity` on the output of Algorithm 13 (which is EFX and 2/3-MMS), we end up with a (partial) EFX allocation which is still 2/3-MMS but also has all the properties that `EFXwithCharity` guarantees.

Theorem 9.14. *For any fair division instance with additive valuations, there exists a (partial)* EFX *allocation $X = (X_1, \ldots, X_n)$ such that*

*(1) X is $\max(2/3, \frac{1}{2-|P(X)|/n})$-*MMS*, and*

(2) for all $i \in [n]$, $v_i(X_i) \geq v_i(P(X))$, and

(3) $|P(X)| < s$,

where s is the number of sources in the envy-graph of X.

Proof. Using Theorem 9.10, we know that there exist a (partial) EFX allocation Y which 2/3-MMS. Then, we can use Theorem 9.13 to obtain a (partial) allocation X with all the stated properties. □

9.4 $\frac{2}{3}$-MMS Together with EF1

In this section, we show that we can compute a complete allocation that is both 2/3-MMS and EF1. Starting from the output of Algorithm 13, we run the well-known *envy-cycle elimination procedure* [55] on the remaining items to obtain an EF1 allocation which is 2/3-MMS as well; see Algorithm 15. We note that our result improves upon the previously best known approximation factor by [7] where they efficiently find allocations that are 4/7-MMS and EF1.

The procedure of envy-cycle elimination was first introduced by [55] that computes an EF1 allocation among agents having monotone valuation; see Algorithm 14 for pseudocode. The idea is to start from an empty allocation and allocate the items one by one such that the partial allocation remains EF1 in each round. In order to do so, one needs to look at the envy-graph of the allocation at each step of the algorithm. If it contains a cycle, by shifting the bundles along that cycle, the utility of all agents on that cycle improves, the allocation remains EF1 and also the number of the edges in the envy-graph decreases. After removing all the cycles, the envy-graph must contain at least one source i.e., an agent whom no one envies. By allocating a remaining item to a source, the allocation remains EF1. While originally, the algorithm starts with an empty

Input: A fair division instance $\mathcal{I} = (\mathcal{N}, \mathcal{M}, \mathcal{V})$ and a partial allocation $X = (X_1, \ldots, X_n, P)$.
Output: A complete allocation $X = (X_1, X_2, \ldots, X_n)$.

while $P \neq \emptyset$ **do**
    while there exists a cycle $i_1 \to i_2 \to \ldots \to i_k \to i_1$ in G_X **do**
        $A \leftarrow X_{i_1}$
        for $j \leftarrow 1$ to $k-1$ **do**
            $X_{i_j} \leftarrow X_{i_{j+1}}$
        $X_{i_k} \leftarrow A$
    Let s be a source in G_X
    Let g be a good in P
    $X_s \leftarrow X_s \cup \{g\}$
    $P \leftarrow P \setminus \{g\}$
return $(X_1, X_2, \ldots, X_n)$

Algorithm 14: `envyCycleElimination`$(\mathcal{I}, X)$

Input: A fair division instance $\mathcal{I} = (\mathcal{N}, \mathcal{M}, \mathcal{V})$.
Output: A complete allocation X.

$X \leftarrow$ `approxMMSandEFX`$(\mathcal{I})$
$X \leftarrow$ `envyCycleElimination`$(\mathcal{I}, X)$ **return** X

Algorithm 15: `approxMMSandEF1`$(\mathcal{I})$

allocation, one can also give a partial allocation as an input to the algorithm and perform the envy-cycle elimination procedure on the input allocation with remaining items. If the input allocation is EF1, then the output allocation will be EF1 as well. See Algorithm 14 for the pseudo-code of our algorithm.

The following lemma follows from the work of [55].

Lemma 9.15. *Given an instance $\mathcal{I}$, if X is a partial* EF1 *allocation, then* `envyCycleElimination`$(\mathcal{I}, X)$ *returns a complete* EF1 *allocation Y in polynomial time such that $v_i(Y_i) \geq v_i(X_i)$ for all $i \in [n]$.*

Proof. [55] showed that `envyCycleElimination`$(\mathcal{I}, X)$ returns a complete EF1 allocation in polynomial time. Fix an agent i. In order to prove that $v_i(Y_i) \geq v_i(X_i)$, it suffices to prove that the value of agent i never decreases throughout the algorithm. Initially, agent i owns X_i. The bundle of agent i only alters if we eliminate a cycle including agent i which in that case agent i receives a bundle which she envied before. Hence her utility increases. Another case is when agent i is the source to whom we allocate a new good. Also in this case the utility of i cannot decrease. Hence, in the end $v_i(Y_i) \geq v_i(X_i)$. □

We now prove our next result that deals with the compatibility of EF1 allocations with MMS guarantees.

Theorem 9.16. *For fair division instances with additive valuations, Algorithm 15 returns a complete allocation which is* EF1 *and* 2/3-MMS.

Proof. For a given fair division instance, Algorithm 15 begins by running Algorithm 13 as a subroutine. By Theorem 9.10, we know that `approxMMSandEFX`($\mathcal{I}$) returns a partial allocation X which is 2/3-MMS and EFX and thus EF1. Then, it runs envy-cycle elimination with the remaining items. And, by Lemma 9.15, we know that `envyCycleElimination`($\mathcal{I}, X$) returns a complete allocation Y which is EF1. Moreover, Lemma 9.15 shows that $v_i(Y_i) \geq v_i(X_i)$ for all agents i. Since X is a 2/3-MMS allocation, Y continues to be a 2/3-MMS allocation as well. This completes our proof. □

Note that, the envy-cycle elimination procedure runs in polynomial time. For any constant $\varepsilon > 0$ and $\delta > 0$, by Theorem 9.11, we can compute a complete a $(2/3 - \varepsilon)$-MMS and EF1 allocation in pseudo-polynomial and by Theorem 9.12, we can compute a $(2/3 - \varepsilon)$-MMS and $(1 - \delta)$-EF1 allocation in polynomial time.

Theorem 9.17. *For fair division instances with additive valuations and any constants $\varepsilon > 0$ and $\delta > 0$, a complete allocation that is both* EF1 *and* $(2/3 - \varepsilon)$-MMS *can be computed in pseudo-polynomial time and a complete allocation that is both* $(1 - \delta)$-EF1 *and* $(2/3 - \varepsilon)$-MMS *can be computed in polynomial time.*

Bibliography

[1] E. Aigner-Horev and E. Segal-Halevi. Envy-free matchings in bipartite graphs and their applications to fair division. *Inf. Sci.*, 587:164–187, 2019.

[2] N. Alon and M. Krivelevich. Divisible subdivisions. *J. Graph Theory*, 98(4):623–629, 2021.

[3] N. Alon and J. Spencer. *The Probabilistic Method*. John Wiley, fourth edition, 2016.

[4] G. Amanatidis, G. Birmpas, A. Filos-Ratsikas, A. Hollender, and A. A. Voudouris. Maximum Nash welfare and other stories about EFX. *Journal of Theoretical Computer Science*, 863:69–85, 2021.

[5] G. Amanatidis, G. Birmpas, and V. Markakis. Comparing approximate relaxations of envy-freeness:42–48, 2018.

[6] G. Amanatidis, E. Markakis, A. Nikzad, and A. Saberi. Approximation algorithms for computing maximin share allocations. *ACM Transactions on Algorithms (TALG)*, 13(4):1–28, 2017. DOI: `10.1145/3147173`.

[7] G. Amanatidis, E. Markakis, and A. Ntokos. Multiple birds with one stone: beating 1/2 for EFX and GMMS via envy cycle elimination. *Theoretical Computer Science*, 841:94–109, 2020.

[8] E. R. Arunachaleswaran, S. Barman, and N. Rathi. Fully polynomial-time approximation schemes for fair rent division. *Mathematics of Operations Research*, 47(3):1970–1998, 2022.

[9] H. Aziz, S. Bouveret, I. Caragiannis, I. Giagkousi, and J. Lang. Knowledge, fairness, and social constraints. In *Proceedings of the 32nd AAAI Conference on Artificial Intelligence (AAAI)*, pages 4638–4645, 2018.

[10] H. Aziz, I. Caragiannis, and A. G. Bogomolnaia. Envy-free up to any item with high Nash welfare: the existence of EFX for binary valuations. *Proceedings of the 2019 ACM Conference on Economics and Computation (EC '19)*, 2019. DOI: `https://doi.org/10.1145/3328526.3329583`.

[11] M. Babaioff, T. Ezra, and U. Feige. Fair-share allocations for agents with arbitrary entitlements. In *Proc. 22nd Conf. Economics and Computation (EC)*, pages 127–127, 2021.

[12] M. Babaioff, T. Ezra, and U. Feige. On best-of-both-worlds fair-share allocations. In *International Conference on Web and Internet Economics*, pages 237–255. Springer, 2022.

[13] S. Barman and S. K. Krishnamurthy. Approximation algorithms for maximin fair division. *ACM Transactions on Economics and Computation (TEAC)*, 8(1):1–28, 2020. DOI: `10.1145/3381525`.

[14] B. A. Berendsohn, S. Boyadzhiyska, and L. Kozma. Fixed-point cycles and approximate EFX allocations. In S. Szeider, R. Ganian, and A. Silva, editors, *47th International Symposium on Mathematical Foundations of Computer Science, MFCS 2022, August 22-26, 2022, Vienna, Austria*, volume 241 of *LIPIcs*, 17:1–17:13. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2022.

[15] B. Berger, A. Cohen, M. Feldman, and A. Fiat. (Almost full) EFX exists for four agents (and beyond). In *Proc. 36th Conf. Artif. Intell. (AAAI)*, 2022.

[16] G. Birkhoff. Tres observaciones sobre el álgebra lineal. Spanish. *Revista de la Universidad de México*, 7(3):18–22, 1940.

[17] S. Bouveret and M. Lemaître. Characterizing conflicts in fair division of indivisible goods using a scale of criteria. *Auton. Agents Multi Agent Syst.*, 30(2):259–290, 2016.

[18] S. J. Brams and A. D. Taylor. *Fair division - from cake-cutting to dispute resolution.* Cambridge University Press, 1996.

[19] X. Bu, Z. Li, S. Liu, J. Song, and B. Tao. Fair division of indivisible goods with comparison-based queries. *arXiv preprint arXiv:2404.18133*, 2024.

[20] E. Budish. The combinatorial assignment problem: approximate competitive equilibrium from equal incomes. *J. Political Economy*, 119(6):1061–1103, 2011.

[21] E. Budish and E. Cantillon. The multi-unit assignment problem: theory and evidence from course allocation at Harvard. *American Economic Review*, 102, 2010.

[22] I. Caragiannis, J. Garg, N. Rathi, E. Sharma, and G. Varricchio. New fairness concepts for allocating indivisible items. In E. Elkind, editor, *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence, IJCAI-23*, pages 2554–2562. International Joint Conferences on Artificial Intelligence Organization, 2023.

[23] I. Caragiannis, N. Gravin, and X. Huang. Envy-freeness up to any item with high Nash welfare: the virtue of donating items. In *Proceedings of the 20th ACM Conference on Economics and Computation (EC)*, pages 527–545, 2019.

[24] I. Caragiannis, D. Kurokawa, H. Moulin, A. D. Procaccia, N. Shah, and J. Wang. Almost envy-free allocations with connected bundles. In *Proceedings of the 2016 ACM Conference on Economics and Computation (EC '16)*, pages 263–279. ACM, 2016. DOI: 10.1145/2940716.2940726.

[25] I. Caragiannis, D. Kurokawa, H. Moulin, A. D. Procaccia, N. Shah, and J. Wang. The unreasonable fairness of maximum nash welfare. In *Proc. 17th Conf. Economics and Computation (EC)*, pages 305–322, 2016.

[26] H. Chan, J. Chen, B. Li, and X. Wu. Maximin-aware allocations of indivisible goods. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 137–143, 2019.

[27] B. R. Chaudhury, J. Garg, and K. Mehlhorn. EFX exists for three agents. In *Proc. 21st Conf. Economics and Computation (EC)*, pages 1–19. ACM, 2020.

[28] B. R. Chaudhury, J. Garg, K. Mehlhorn, R. Mehta, and P. Misra. Improving EFX guarantees through rainbow cycle number. In *Proceedings of the 22nd ACM Conference on Economics and Computation (EC)*, pages 310–311. ACM, 2021.

[29] B. R. Chaudhury, T. Kavitha, K. Mehlhorn, and A. Sgouritsa. A little charity guarantees almost envy-freeness. In *Proc. 31st Symp. Discrete Algorithms (SODA)*, pages 2658–2672, 2020.

[30] B. R. Chaudhury, T. Kavitha, K. Mehlhorn, and A. Sgouritsa. A little charity guarantees almost envy-freeness. *SIAM Journal on Computing*, 50(4):1336–1358, 2021. DOI: 10.1137/20M1359134.

[31] X. Deng, Q. Qi, and A. Saberi. Algorithmic solutions for envy-free cake cutting. *Operations Research*, 60(6):1461–1476, 2012.

[32] B. L. Deuermeyer, D. K. Friesen, and M. A. Langston. Scheduling to maximize the minimum processor finish time in a multiprocessor system. *SIAM Journal on Algebraic Discrete Methods*, 3(2):190–196, 1982. DOI: 10.1137/0603019.

[33] L. E. Dubins and E. H. Spanier. How to cut a cake fairly. *The American Mathematical Monthly*, 68(1P1):1–17, 1961.

[34] R. Etkin, A. Parekh, and D. Tse. Spectrum sharing for unlicensed bands. In *In Proceedings of the first IEEE Symposium on New Frontiers in Dynamic Spectrum Access Networks*, 2005.

[35] A. Farhadi, M. Hajiaghayi, M. Latifian, M. Seddighin, and H. Yami. Almost envy-freeness, envy-rank, and Nash social welfare matchings. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35 of number 6, pages 5355–5362, 2021.

[36] U. Feige and A. Norkin. Improved maximin fair allocation of indivisible items to three agents, 2022. DOI: 10.48550/ARXIV.2205.05363.

[37] U. Feige, A. Sapir, and L. Tauber. A tight negative example for MMS fair allocations. In *Proc. 17th Conf. Web and Internet Economics (WINE)*, pages 355–372, 2021.

[38] M. Feldman, S. Mauras, V. V. Narayan, and T. Ponitka. Breaking the envy cycle: best-of-both-worlds guarantees for subadditive valuations. *CoRR*, abs/2304.03706, 2023. DOI: 10.48550/arXiv.2304.03706. arXiv: 2304.03706. URL: https://doi.org/10.48550/arXiv.2304.03706.

[39] R. Freeman, N. Shah, and R. Vaish. Best of both worlds: ex-ante and ex-post fairness in resource allocation. In *Proceedings of the 21st ACM Conference on Economics and Computation*, pages 21–22, 2020.

[40] J. Gan, W. Suksompong, and A. A. Voudouris. Envy-freeness in house allocation problems. *Mathematical Social Sciences*, 101:104–106, 2019. ISSN: 0165-4896. DOI: https://doi.org/10.1016/j.mathsocsci.2019.07.005. URL: https://www.sciencedirect.com/science/article/pii/S0165489619300599.

[41] J. Garg, P. McGlaughlin, and S. Taki. Approximating maximin share allocations. In *Symposium on Simplicity in Algorithms (SOSA 2019)*, volume 69, 20:1–20:11, 2018. DOI: 10.4230/OASIcs.SOSA.2019.20.

[42] J. Garg and S. Taki. An improved approximation algorithm for maximin shares. *Artificial Intelligence*:103547, 2021.

[43] V. Gates, T. L. Griffiths, and A. D. Dragan. How to be helpful to multiple people at once. *Cognitive Science*, 44, 2020.

[44] M. Ghodsi, M. HajiAghayi, M. Seddighin, S. Seddighin, and H. Yami. Fair allocation of indivisible goods: improvements and generalizations. In *Proc. 19th Conf. Economics and Computation (EC)*, pages 539–556, 2018.

[45] P. W. Goldberg, K. Høgh, and A. Hollender. The frontier of intractability for EFX with two agents. In *Algorithmic Game Theory: 16th International Symposium, SAGT 2023, Egham, UK, September 4–7, 2023, Proceedings*, 290–307. Springer-Verlag, 2023.

[46] H. Hosseini and A. Searns. Guaranteeing maximin shares: some agents left behind. In *Proc. 30th Intl. Joint Conf. Artif. Intell. (IJCAI)*, pages 238–244, 2021.

[47] H. Hosseini, A. Searns, and E. Segal-Halevi. Ordinal maximin share approximation for goods. *J. Artif. Intell. Res.*, 74, 2021.

[48] H. Hummel. Maximin shares in hereditary set systems, 2024. arXiv: 2404.11582 [cs.GT]. URL: https://arxiv.org/abs/2404.11582.

[49] S. C. Jahan, M. Seddighin, S. M. S. Javadi, and M. Sharifi. Rainbow cycle number and EFX allocations: (almost) closing the gap. In *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence, IJCAI 2023, 19th-25th August 2023, Macao, SAR, China*, pages 2572–2580. ijcai.org, 2023.

[50] D. S. Johnson, C. H. Papadimitriou, and M. Yannakakis. How easy is local search? *Journal of computer and system sciences*, 37(1):79–100, 1988.

[51] H. W. Kuhn. *Chapter 2. on games of fair division.* In *Essays in Mathematical Economics, in Honor of Oskar Morgenstern.* Princeton University Press, Princeton, 1967, pages 29–38. ISBN: 9781400877386. DOI: doi:10.1515/9781400877386-004. URL: https://doi.org/10.1515/9781400877386-004.

[52] D. Kurokawa, A. D. Procaccia, and J. Wang. Fair enough: guaranteeing approximate maximin shares. *J. ACM*, 65(2):8:1–8:27, 2018.

[53] D. Kurokawa, A. D. Procaccia, and J. Wang. Fair enough: guaranteeing approximate maximin shares. *Journal of the ACM (JACM)*, 65(2):1–27, 2018. DOI: 10.1145/3140756.

[54] D. Kurokawa, A. D. Procaccia, and J. Wang. When can the maximin share guarantee be guaranteed? In *Proc. 30th Conf. Artif. Intell. (AAAI)*, pages 523–529, 2016.

[55] R. J. Lipton, E. Markakis, E. Mossel, and A. Saberi. On approximately fair allocations of indivisible goods. In *Proceedings of the 5th ACM conference on Electronic commerce*, pages 125–131. ACM, 2004.

[56] R. Mahara. Extension of additive valuations to general valuations on the existence of EFX. In *ESA*, volume 204 of *LIPIcs*, 66:1–66:15. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2021.

[57] T. Mészáros and R. Steiner. Zero sum cycles in complete digraphs. *Eur. J. Comb.*, 98:103399, 2021.

[58] H. Moulin. Fair division in the internet age. *Annual Review of Economics*, 11(1):407–441, 2019.

[59] B. Plaut and T. Roughgarden. Almost envy-freeness with general valuations. *SIAM J. Discret. Math.*, 34(2):1039–1068, 2020.

[60] J. W. Pratt and R. J. Zeckhauser. The fair and efficient division of the Winsor family silver. *Management Science*, 36(11):1293–1301, 1990.

[61] A. D. Procaccia. Technical perspective: an answer to fair division's most enigmatic question. *Communications of the ACM*, 63(4):118–118, 2020.

[62] A. D. Procaccia and J. Wang. Fair enough: guaranteeing approximate maximin shares. In *Proc. 15th Conf. Economics and Computation (EC)*, pages 675–692, 2014.

[63] M. Seddighin and S. Seddighin. Improved maximin guarantees for subadditive and fractionally subadditive fair allocation problem. In *AAAI 2022, IAAI 2022, EAAI 2022 Virtual Event, February 22 - March 1, 2022.* AAAI Press, 2022.

[64] H. Steinhaus. The problem of fair division. *Econometrica*, 16:101–104, 1948.

[65] W. Stromquist. Envy-free cake divisions cannot be found by finite protocols. *the electronic journal of combinatorics*, 15(1):11, 2008.

[66] W. Stromquist. How to cut a cake fairly. *The American Mathematical Monthly*, 87(8):640–644, 1980.

[67] F. E. Su. Rental harmony: sperner's lemma in fair division. *American Mathematical Monthly*, 106:930–942, 1999.

[68] G. B. Uziahu and U. Feige. On fair allocation of indivisible goods to submodular agents. *CoRR*, abs/2303.12444, 2023. DOI: `10.48550/ARXIV.2303.12444`. arXiv: `2303.12444`. URL: `https://doi.org/10.48550/arXiv.2303.12444`.

[69] J. von Neumann. A contribution to the theory of games. *Annals of Mathematics*, 41(4):296–320, 1940. DOI: `10.2307/1968907`.

[70] T. W. Vossen. *Fair allocation concepts in air traffic management.* PhD thesis, University of Maryland, College Park, 2002.

[71] G. J. Woeginger. A polynomial-time approximation scheme for maximizing the minimum machine completion time. *Oper. Res. Lett.*, 20(4):149–154, 1997.