Maximin Fairness with Mixed Divisible and Indivisible Goods

Xiaohui Bei,[1] Shengxin Liu,[2] * Xinhang Lu,[1] Hongao Wang[1]
[1] Nanyang Technological University, Singapore
[2] Harbin Institute of Technology, Shenzhen, China
xhbei@ntu.edu.sg, sxliu@hit.edu.cn, {xinhang001@e., hongao.wang@}ntu.edu.sg

Abstract

We study fair resource allocation when the resources contain a mixture of divisible and indivisible goods, focusing on the well-studied fairness notion of maximin share fairness (MMS). With only indivisible goods, a full MMS allocation may not exist, but a constant multiplicative approximate allocation always does. We analyze how the MMS approximation guarantee would be affected when the resources to be allocated also contain divisible goods. In particular, we show that the worst-case MMS approximation guarantee with mixed goods is no worse than that with only indivisible goods. However, there exist problem instances to which adding some divisible resources would strictly decrease the MMS approximation ratios of the instances. On the algorithmic front, we propose a constructive algorithm that will always produce an α-MMS allocation for any number of agents, where α takes values between $1/2$ and 1 and is a monotonically increasing function determined by how agents value the divisible goods relative to their MMS values.

1 Introduction

Fair division concerns the problem of allocating a set of goods among interested agents in a way that is *fair* to all participants involved. The goods involved could be heterogeneous and *divisible*, usually modelled by a *cake*, in which case the problem is also known as *cake-cutting*; in some other cases, the goods are heterogeneous and *indivisible*, and the problem is known as *indivisible resource allocation*.

Due to its subjective nature, a plethora of fairness notions have been proposed and investigated in different resource allocation scenarios (see (Young 1995) and (Brams and Taylor 1996) for a survey). In particular, as one of the most classic and widely known fairness notions, Steinhaus (1948) proposed that in an allocation that involves n participating agents, each agent should receive a bundle which is worth at least $1/n$ of her value for the entire set of goods. An allocation satisfying such property is then known as a *proportional* allocation. Moreover, Steinhaus (1948) also showed that a proportional allocation can always be found for any number of agents over any divisible good. However, this is not the case when goods are *indivisible*, with the simplest counterexample of two agents dividing a single valuable good. In order to circumvent this issue, Budish (2011) presented a natural alternative to the classic proportionality notion that also works for indivisible goods, known as the *maximin share (MMS) guarantee*. In this definition, the *maximin share (MMS)* of an agent is defined as the largest value she can get if she is allowed to partition goods into n bundles and always receives the least desirable bundle. An allocation is said to be an *MMS allocation* if every agent receives a bundle which is worth at least her maximin share.

The notion of MMS nicely captures the local measure of fairness even when the goods to be allocated are indivisible. A natural question then arises of whether an MMS allocation always exists in all problem instances. Surprisingly, Kurokawa, Procaccia, and Wang (2018) showed that even with additive valuation functions, an MMS allocation may not always exist. However, a $2/3$-MMS allocation can always be found, in which each agent is guaranteed to receive a bundle worth at least $2/3$ of their MMS value. In other words, if we define the *MMS approximation guarantee* of a problem instance as the largest α such that the instance admits an α-MMS allocation, the results in (Kurokawa, Procaccia, and Wang 2018) imply that the worst MMS approximation guarantee across all indivisible problem instances is strictly less than 1 and at least 2/3. Since then, many subsequent works have been carried out on the improvements of MMS approximation guarantee, design of simpler algorithms, etc. (Amanatidis et al. 2017; Barman and Krishnamurthy 2020; Ghodsi et al. 2018; Garg, McGlaughlin, and Taki 2019; Garg and Taki 2020). MMS has also been adopted as the fairness solution concept in several practical applications (Budish 2011; Goldman and Procaccia 2015).

Even though MMS has been mainly studied in the context of indivisible resource allocation, it is also a well-defined fairness notion in a more general setting where both divisible and indivisible goods are to be allocated. Many real-world scenarios, including but not limited to divorce or inheritance settlements, involve allocating simultaneously divisible goods such as land or money and indivisible goods such as houses or cars. What fairness notion should one adopt when dividing resources of such mixed types? The problem of fairly allocating mixed divisible and indivisible goods was first studied by Bei et al. (2020a) (and recently by Bhaskar,

*Shengxin Liu is the corresponding author.

Sricharan, and Vaish (2020)), in which the authors proposed a new fairness notion called *envy-freeness for mixed goods (EFM)* that generalizes envy-freeness, another well-studied fairness notion, to the mixed goods setting. The maximin share guarantee, on the other hand, can be directly applied to the mixed goods setting without any modification. This allows us to compare the results of MMS for mixed goods directly to those for indivisible goods.

In this paper, we aim to provide such a comparison. More specifically, we extend the analysis of MMS allocations to the setting with mixed types of goods, and study its existence, approximation, as well as computation. In particular, we hope to answer the following questions:

1. Is the worst-case MMS approximation guarantee across all mixed goods instances the same as that across all indivisible goods instances?
2. Given any problem instance, would adding some divisible resources to it always (weakly) increase the MMS approximation ratio of this instance?
3. How to design algorithms that could find allocations with good MMS approximation guarantee in mixed goods problem instances?

1.1 Our Results

In this paper, we answer the three questions posed above.

In Section 3, we first show that any problem instance of mixed goods can be converted into another problem instance with only indivisible goods, such that the two instances have the same MMS value for every agent, and any allocation of the indivisible instance can be converted to an allocation in the mixed instance. This reduction directly implies that the worst-case MMS approximation guarantee across all mixed goods instances is the same as that across all indivisible goods instances.

This is not a surprising result, because the non-existence of MMS allocations only arises when the resources to be allocated become indivisible. It is therefore reasonable to think that adding divisible goods to the set of indivisible goods can only help with the MMS approximation guarantee. However, we show that this intuition no longer holds at the per-instance level. In particular, we provide a problem instance with only indivisible goods, such that when a small amount of divisible goods is added to the instance, the MMS approximation guarantee of the instance *strictly decreases*, i.e., while an α-MMS allocation exists in the original instance, no α-MMS allocation exists after adding cake.

Next in Section 4, we focus on finding allocations with good MMS approximations with mixed types of goods. More specifically, we show via a constructive algorithm that given any problem instance with mixed goods, there exists an α-MMS allocation, where the parameter α, ranged between $1/2$ and 1, is a monotonically increasing function of how agents value the divisible goods relative to their MMS values. This means when agents have more divisible goods with them, one can achieve a better MMS approximation guarantee. The idea of the algorithm is to repeatedly assign some agent a set of indivisible goods along with a piece of cake to reach the agent's α-MMS value, and then reduce the problem to a smaller size. When the cake to be allocated is heterogeneous, the algorithm also makes use of a generalized fairness notion of *weighted proportionality* to help allocate the cake. On the computational front, we show polynomial-time approximation schemes for approximating the MMS value of an agent and for computing a $(1-\epsilon)\alpha$-MMS allocation in a mixed goods problem instance. These algorithms run in time polynomial in n, m, L for any constant $\epsilon > 0$, where n is the number of agents, m is the number of indivisible goods, L is the input bit length.

Last, in Section 5, we discuss the relation between MMS and the recently introduced envy-freeness for mixed goods (EFM) in the mixed goods setting. Generally speaking, neither the MMS nor the EFM imply the other. We also provide a result showing what fraction of MMS can be implied by an EFM allocation.

1.2 Further Related Work

Maximin share (MMS) fairness was first introduced by Budish (2011). In addition to the works we mentioned above, MMS allocations of indivisible resources have also been extensively studied in several other settings (Farhadi et al. 2019; Suksompong 2018; Bouveret et al. 2017; Gourvès and Monnot 2019; Igarashi and Peters 2019; Lonc and Truszczynski 2020).

A related line of research incorporates money into the fair division of indivisible goods, with the consideration of finding envy-free allocations (Alkan, Demange, and Gale 1991; Maskin 1987; Klijn 2000; Meertens, Potters, and Reijnierse 2002; Halpern and Shah 2019; Brustle et al. 2020; Caragiannis and Ioannidis 2020; Aziz 2021). Another closely related problem is *rent division* (see (Su 1999; Haake, Raith, and Su 2002; Abdulkadiroğlu, Sönmez, and Ünver 2004; Brams 2008; Gal et al. 2017; Arunachaleswaran, Barman, and Rathi 2019)). Its cardinal utility version can be viewed as a special case of the mixed setting where one wants to allocate (indivisible) rooms and the (divisible) rent among agents. However, in the mixed setting of fair division, the divisible goods (the rent) must be allocated and the agents are not allowed to use additional money to achieve more strict fairness condition.

2 Preliminaries

Denote by $N = \{1, 2, \dots, n\}$ the set of agents. Let $M = \{1, 2, \dots, m\}$ be the set of indivisible goods. Each agent $i \in N$ has a non-negative utility $u_i(g)$ for each indivisible good $g \in M$. We assume that each agent's utility for a set of indivisible goods is additive, that is, $u_i(M') = \sum_{g \in M'} u_i(g)$ for any $i \in N$ and $M' \subseteq M$. Let $C = \{D_1, D_2, \dots, D_\ell\}$ be the set of heterogeneous divisible goods. We assume without loss of generality that each cake $D_i \in C$ is denoted by the interval $[(i-1)/\ell, i/\ell]$. Thus the entire set of divisible goods is represented by one cake $C = [0, 1]$.[1] A *piece* of cake is a *finite* union of subintervals of $[0, 1]$. Each agent i has a non-negative integrable density function f_i. Given a piece

[1]The agents' density functions over the cakes are assumed to be non-atomic. This property allows us to view two consecutive intervals as disjoint if their intersection is a singleton.

of cake $S \subseteq [0,1]$, agent i's value over S is then defined as $u_i(S) := \int_{x \in S} f_i(x)\ dx$. Denote by $\mathbf{f} = (f_1, f_2, \ldots, f_n)$ the vector of density functions; $\mathbf{f}$ is called a *density profile*. In this work, a resource allocation *problem instance* $I = \langle N, M \cup C \rangle$ consists of a set of agents N (together with their utility and density functions), a set of indivisible goods M, and a set of heterogeneous divisible goods or *cakes* C.

Denote by $G = M \cup C$ the set of *mixed* goods (by abuse of notation for convenience). Let $\mathcal{M} = (M_1, M_2, \ldots, M_n)$ be a partition of indivisible goods M into n bundles such that agent i receives M_i. Let $\mathcal{C} = (C_1, C_2, \ldots, C_n)$ be a partition of the cake C such that agent i gets a piece of cake C_i. An *allocation* of mixed goods $G = M \cup C$ is defined as $\mathcal{A} = (A_1, A_2, \ldots, A_n)$, where $A_i = M_i \cup C_i$ is allocated to agent i. The *utility* of agent i in an allocation $\mathcal{A}$ is then $u_i(A_i) = u_i(M_i) + u_i(C_i)$.

We now define the fairness notions considered in this paper. We focus on the *maximin share fairness*, a generalization of the classic *proportionality fairness*.

Definition 2.1 (PROP)**.** An allocation $\mathcal{A}$ is said to satisfy *proportionality (PROP)* if for each agent $i \in N$, $u_i(A_i) \geq u_i(G)/n$.

Definition 2.2. Let $\Pi_k(G) = \{\{P_1, P_2, \ldots, P_k\} \mid P_i \cap P_j = \emptyset\ \forall i,j \text{ and } \cup_k P_k = G\}$ be the set of k-partitions of G. Define the k-maximin share of agent i as

$$\mathrm{MMS}_i(k, G) = \max_{\mathcal{P}=(P_1,P_2,\ldots,P_k)\in\Pi_k(G)} \min_{j\in[k]} u_i(P_j).$$

The *maximin share* of agent i is $\mathrm{MMS}_i(n, G)$. Every partition in $\arg\max_{\mathcal{P}\in\Pi_n(G)} \min_{j\in[n]} u_i(P_j)$ is called an *MMS partition* for agent i.

For notational convenience, we will simply write MMS_i when parameters n and G are clear from the context.

Definition 2.3 (α-MMS)**.** An allocation $\mathcal{A}$ of mixed goods G is said to satisfy the *α-approximate maximin share fairness (α-MMS)*, for some $\alpha \in [0,1]$, if for every agent $i \in N$,

$$u_i(A_i) \geq \alpha \cdot \mathrm{MMS}_i(n, G).$$

We say a 1-MMS (or full-MMS) allocation satisfies the (full) maximin share fairness and write MMS as a shorthand for 1-MMS. To slightly abuse the notation, we will also refer to an agent's maximin share as MMS.

Precision and input representation. When discussing the computational aspects, it is necessary to specify the precision and representation of the input problem instance. In this paper, we assume that $u_i(g)$'s for each $i \in N, g \in M$ and $u_i(C)$ for each $i \in N$ are all rational numbers, and the whole input can be represented in no more than L bits.

Robertson-Webb query model. We also adopt the Robertson-Webb (RW) query model to access agents' density functions for the cake. In the RW model, an algorithm is allowed to ask each agent the following two types of queries:

Eval: An evaluation query returns $u_i([x,y])$ of agent i over interval $[x,y]$.

Cut: A cut query of β for agent i from point x returns a point y such that $u_i([x,y]) = \beta$.

In this paper, we assume that each query in the RW model takes unit time.

All omitted proofs can be found in the full version of this paper (Bei et al. 2020b).

3 MMS Approximation Guarantee

In this section, we examine how mixed goods affect the existence and approximation of MMS allocations.

3.1 Worst Case MMS Approximation Guarantee

An MMS allocation, while being an appealing solution concept, may not always exist in every problem instance with indivisible goods (Kurokawa, Procaccia, and Wang 2018). Therefore one has to resort to approximate MMS allocations. Allocating mixed types of goods is a generalization of the indivisible good case, and hence suffers from the same issue. We start by analyzing the worst-case MMS approximation guarantee for mixed good problem instances.

Definition 3.1. Given a mixed good problem instance I, let $\gamma(I)$ denote the maximum value of α such that the problem instance admits an α-MMS allocation. [2] We also call $\gamma(I)$ the *MMS approximation guarantee* of problem instance I.

We further define two constants

$$\gamma_M = \inf_{I=\langle N, M\cup C\rangle} \gamma(I) \quad \text{and} \quad \gamma_I = \inf_{I=\langle N, M\rangle} \gamma(I).$$

In other words, γ_M is the worst MMS approximation guarantee across all mixed goods problem instances, and γ_I is the worst MMS approximation guarantee across all problem instances that contain only indivisible goods. Previous works have showed that $\gamma_I < 1$ (Kurokawa, Procaccia, and Wang 2018) and $\gamma_I \geq \frac{3}{4} + \frac{1}{12n}$ (Garg and Taki 2020).

It is straightforward from definition that $\gamma_M \leq \gamma_I$. In the following, our first result shows that γ_M is also no less than γ_I. This is proved via the following reduction theorem.

Theorem 3.2. *Given any problem instance with mixed goods $I = \langle N, M \cup C \rangle$, there exists another problem instance $I' = \langle N, M' \rangle$ with only indivisible items M' and the same set N of agents, such that*

- *any allocation $\mathcal{A}'$ of M' can be converted to another allocation $\mathcal{A}$ of $M \cup C$, such that $u_i(A_i) = u_i(A'_i)$ for each agent $i \in N$;*
- *$\mathrm{MMS}_i(n, M \cup C) = \mathrm{MMS}_i(n, M')$ for each agent $i \in N$.*

We note that this reduction is not computationally efficient as it requires being able to compute the MMS values. Moreover, Theorem 3.2 directly implies the following result.

Corollary 3.3. $\gamma_I = \gamma_M$.

In other words, having mixed types of goods does not affect the worst-case MMS approximation guarantee across all problem instances. As another corollary, this also means that if there exists a universal β-MMS algorithm for indivisible goods for some β, it immediately implies that every problem instance of mixed goods also admits a β-MMS allocation. We will discuss more on the algorithmic implication of this result in Section 4.

[2]The $\gamma(I)$ is defined to be the maximum value of α instead of the supremum. This is because the density functions are non-atomic and the maximum α can always be achieved.

3.2 Cake Does Not Always Help

Note that the equation in Corollary 3.3 is about the worst-case MMS approximation guarantee across all problem instances. Next we show that such equivalence may not hold on a per-instance level. In particular, we will demonstrate via an example that sometimes, adding some divisible resources to some problem instance I may hurt its MMS approximation guarantee value $\gamma(I)$.

Theorem 3.4. *There exist some agent set N, indivisible goods M, and divisible goods C, such that*

$$\gamma(\langle N, M\rangle) > \gamma(\langle N, M \cup C\rangle).$$

In other words, adding some divisible goods to the set of resources may decrease the MMS approximation guarantee of this problem instance in some cases.

In the following we explain the intuition of the theorem proof. We want to find a problem instance $I = \langle N, M\rangle$ such that $\gamma(\langle N, M\rangle) < 1$, and the instance should have the following properties.

Fix an agent i. In her MMS partition, the least valued bundle is unique, i.e., the value of the least valued bundle is *strictly* less than that of the second least valued bundle. If this is the case, then given a cake C with a small enough value ϵ, the new MMS value $\text{MMS}_i(n, M \cup C)$ should be exactly $\text{MMS}_i(n, M) + \epsilon$. Now suppose that in the instance I, all the agents have this property. This means that every agent's MMS value will increase by ϵ when we add a cake C of a small enough value ϵ to the instance I. The second required property of I is that in any $\gamma(\langle N, M\rangle)$-MMS allocation, there are at least two agents that receive exactly $\gamma(\langle N, M\rangle)$ times their MMS values.

With these two properties, the actual cake C will not be enough for distributing to all the agents while clinging to a large enough MMS approximation ratio $\gamma(\langle N, M \cup C\rangle)$. In other words, with the cake C added, the new MMS ratio $\gamma(\langle N, M \cup C\rangle)$ will decrease, comparing to $\gamma(\langle N, M\rangle)$.

Finally, the counterexample used to show the non-existence of MMS allocation in (Kurokawa, Procaccia, and Wang 2018) can be utilized to construct the instance I that satisfies all above mentioned properties. The details are omitted here.

4 Algorithms for Computing Approximate MMS Allocations

The previous section investigates MMS approximation guarantee, which is the *best possible* MMS approximation of a problem instance. In this section, our goal is to design algorithms that could compute allocations with *good* MMS approximation ratios in a mixed goods problem instance. We hope such an algorithm can be flexible, in the sense that when the problem instance contains only indivisible goods, the MMS approximation of the output allocation should match or be close to the previously best-known approximation ratio for indivisible goods; on the other hand, when the resources contain enough divisible goods, the indivisible goods would become negligible, and our algorithm should be able to produce an allocation that gives each agent their full MMS value.

As the main result of this section, in the following we present such an algorithm. We will show that the algorithm will always produce an α-MMS allocation in the mixed goods setting, where α is a monotonically increasing function of how agents value the divisible goods relative to their MMS values and ranges between $1/2$ and 1.

Theorem 4.1. *Given any mixed good problem instance $\langle N, M \cup C\rangle$, an α-MMS allocation always exists, where*

$$\alpha = \min\left\{1, \frac{1}{2} + \min_{i \in N}\left\{\frac{u_i(C)}{2(n-1) \cdot \mathit{MMS}_i}\right\}\right\}.$$

Furthermore, for any constant $\epsilon > 0$, we can compute a ratio α' and an allocation $\mathcal{A}$ in time polynomial in n, m, L such that:

1. *$\alpha' \geq \alpha$, and*
2. *the allocation $\mathcal{A}$ is $(1-\epsilon)\alpha'$-MMS.*

Here n is the number of agents, m is the number of items, and L is the total bit length of all input parameters.

Theorem 4.1 has several implications. For example, when every agent i has $u_i(C) \geq (n/2)\text{MMS}_i$, Theorem 4.1 implies the existence of an α-MMS allocation with α better than the currently best-known approximation ratio $\frac{3}{4} + \frac{1}{12n}$ with indivisible goods due to Garg and Taki (2020). In addition, the following corollary shows the amount of divisible good needed to ensure that the instance admits a full-MMS allocation.

Corollary 4.2. *Given a mixed good problem instance $I = (N, M \cup C)$, if $u_i(C) \geq (n-1)\mathit{MMS}_i$ holds for each agent $i \in N$, then an MMS allocation is guaranteed to exist.*

This means even with the presence of indivisible items, as long as there are enough divisible goods, a full-MMS allocation can always be found. However, this corollary should not be interpreted as that this is the *least* amount of divisible goods required. For example, Halpern and Shah (2019) and Brustle et al. (2020) studied the allocation of indivisible goods and a very special type of divisible goods, money. They investigated the least amount of money needed for a problem instance to have an envy-free allocation. Although an envy-free allocation is also a full-MMS allocation, their result and this corollary are incomparable.

The remaining of this section is dedicated to the proof of Theorem 4.1. The proof consists of the following steps.

Section 4.1: We first focus on a restricted case in which the cake to be allocated is *homogeneous* to every agent. We show via a constructive, but not necessarily polynomial time algorithm, that an α-MMS allocation always exists in this setting.

Section 4.2: Next we generalize the above algorithm to the general case with heterogeneous cake, using the concept of *weighted proportionality* in cake-cutting.

Section 4.3: We discuss how to convert the algorithm to a polynomial time algorithm at the cost of a small loss in the MMS approximation ratio.

We also discuss how to further improve the approximation ratio α in Section 4.4.

4.1 Homogeneous Cake

We begin with a special case where the cake to be allocated is *homogeneous*, meaning that each agent values all pieces of equal size the same. In other words, the value of a piece of cake to each agent depends only on the length of the piece.[3] We refer to the homogeneous cake as $\hat{C}$. Formally, given a piece of homogeneous cake $S \subseteq [0,1]$, each agent i's value over S is then defined as $u_i(S) := (\sum_{[a,b]\in S}(b-a))u_i(\hat{C})$.

The Algorithm The complete algorithm to compute an α-MMS allocation is shown in Algorithm 1. Our algorithm is in spirit similar to the algorithm in (Ghodsi et al. 2018). After initialization, the algorithm can be decomposed into two phases as follows:

- Phase 1: allocate big goods (lines 4-6). Algorithm 1 repeatedly allocates some agent a single indivisible good which has value at least α times this agent's MMS value. Then, both the agent and the allocated good are removed from all further considerations.
- Phase 2: allocate small goods (lines 7-13). This phase executes in rounds. In each round, Algorithm 1 chooses an agent i^* and allocates some indivisible goods B (formed at line 9) along with a piece of cake $[a, x_{i^*}]$ to agent i^* (line 12). Then again, both the agent and her goods are removed from the instance.

The Analysis Algorithm 1 consists of two phases. We analyze each of them separately.

Phase 1: Allocate big goods. First, when goods are all indivisible, Amanatidis et al. (2017) showed that allocating a single good to an agent does not decrease the MMS values of other agents. Here we show that this result holds in the mixed goods setting as well.

Lemma 4.3 (Monotonicity property)**.** *Given an instance $(N, G = M \cup C)$, for any agent $i \in N$ and any indivisible good $g \in M$, it holds that $MMS_i(n-1, G \setminus \{g\}) \geq MMS_i(n, G)$.*

Denote by N_1 the set of remaining agents and G_1 the set of unallocated goods just before Phase 2 is executed. Let $n_1 = |N_1|$. Applying the monotonicity property (Lemma 4.3) $n - n_1$ times, we have that for each agent $i \in N_1$, $\text{MMS}_i(n_1, G_1) \geq \text{MMS}_i(n, G)$. In addition, each agent i who leaves the system in this phase receives an item of value at least $\alpha \cdot \text{MMS}_i$. This implies that Phase 1 will not affect the correctness and termination of Algorithm 1. It simply adds the property that in Phase 2, each remaining agent i will value each of the remaining indivisible goods less than $\alpha \cdot \text{MMS}_i$.

Phase 2: Allocate small goods. In this phase, at each round, for the agent i^* selected at line 11, we show that it satisfies two properties:

(1) $u_{i^*}(A_{i^*}) \geq \alpha \cdot \text{MMS}_{i^*}$;

(2) For each agent j remaining in N, $u_j(A_{i^*}) \leq \text{MMS}_j$.

[3]An even more restricted case is when the cake is valued the same to all agents. The canonical example of the divisible goods of this special case is *money*.

Algorithm 1: MIXED-MMS-HOMOGENEOUS($\langle N, M \cup \hat{C}\rangle$)

Input: Agents N, indivisible goods M and a homogeneous cake $\hat{C}$, utility and density functions.

1 Compute MMS_i, for each $i \in N$.

2 $\alpha \leftarrow \min\left\{1, \frac{1}{2} + \min_{i\in N}\left\{\frac{u_i(\hat{C})}{2(n-1)\cdot \text{MMS}_i}\right\}\right\}$

3 $A_1, A_2, \ldots, A_n \leftarrow \emptyset$

`// Phase 1: allocate big goods.`

4 **while** $\exists i \in N, g \in M$ *such that* $u_i(g) \geq \alpha \cdot MMS_i$ **do**

5     $A_i \leftarrow \{g\}$ `// arbitrary tie-breaking`

6     $N \leftarrow N \setminus \{i\}, M \leftarrow M \setminus \{g\}$

`// Phase 2: allocate small goods.`

7 **while** $|N| \geq 2$ **do**

8     $B \leftarrow \emptyset$

9   Add one indivisible good at a time to B until $u_j(B) \geq (1-\alpha) \cdot \text{MMS}_j$ for some agent j or $B = M$.

10   Suppose $\hat{C} = [a, b]$. For each $i \in N$, let x_i be the leftmost point with $u_i(B \cup [a, x_i]) \geq \alpha \cdot \text{MMS}_i$.

11     $i^* \leftarrow \arg\min_{i\in N} x_i$ `// arbitrary tie-breaking`

12     $A_{i^*} \leftarrow B \cup [a, x_{i^*}]$

13     $N \leftarrow N \setminus \{i^*\}, M \leftarrow M \setminus B, \hat{C} \leftarrow \hat{C} \setminus [a, x_{i^*}]$

14 Give all remaining goods to the last agent.

15 **return** $(A_1, A_2, \ldots, A_n)$

---

(1) is straightforward by the way each x_i is computed at line 10. To show (2) is true, we remark that no single good is valued more than $\alpha \cdot \text{MMS}_i$ for any agent i. Therefore, the set B selected at line 9 must satisfy $u_j(B) \leq \text{MMS}_j$ for all $j \in N$. After line 10, it continues to satisfy that $u_j(B \cup [a, x_j]) \leq \text{MMS}_j$ for each $j \in N$. Then, because i^* is selected such that x_{i^*} is the smallest value, one would have $u_j(A_{i^*} = B \cup [a, x_{i^*}]) \leq u_j(B \cup [a, x_j]) \leq \text{MMS}_j$ for each agent $j \in N$.

In particular, property (2) ensures that the last agent at line 14 is still left with enough goods to reach her maximin share. Therefore, every agent i will receive value at least $\alpha \cdot \text{MMS}_i$ after the two phases. It only remains to show that the cake $\hat{C}$ is enough to be allocated throughout the process.

Lemma 4.4. *Cake $\hat{C}$ is enough to be allocated in Algorithm 1. In other words, x_i for each agent $i \in N$ at line 10 is always well defined in each round.*

Combining everything together, we conclude that Algorithm 1 is a correct algorithm that always outputs an α-MMS allocation.

4.2 Heterogeneous Cake

We now show how to extend algorithm 1 to the general setting with a heterogeneous cake C. The new algorithm follows a very simple idea as follows. First we replace cake C

Algorithm 2: The Mixed MMS Algorithm

Input: Agents N, indivisible goods M and cake C, utility and density functions.

1 Let $\hat{C} = [0, 1]$ be a homogeneous cake with $u_i(\hat{C}) = u_i(C)$ for each agent $i \in N$.
2 $(M_1 \cup \hat{C}_1, M_2 \cup \hat{C}_2, \ldots, M_n \cup \hat{C}_n) \leftarrow$ MIXED-MMS-HOMOGENEOUS$(\langle N, M \cup \hat{C} \rangle)$
3 For each $i \in N$, let $w_i \leftarrow u_i(\hat{C}_i)/u_i(C)$ if $u_i(C) > 0$; $w_i \leftarrow 0$ otherwise.
4 $(C_1, C_2, \ldots, C_n) \leftarrow$ WPRALLOC$(N, C, \mathbf{w} = (w_1, \ldots, w_n))$ `// allocate cake` C
5 **return** $(M_1 \cup C_1, M_2 \cup C_2, \ldots, M_n \cup C_n)$

with a homogeneous cake $\hat{C}$ such that $u_i(\hat{C}) = u_i(C)$ for each agent i, and allocate resources M and $\hat{C}$ to all agents using Algorithm 1. Let $\hat{C}_i$ be the piece allocated to agent i. Note that since $\hat{C}$ is homogeneous, only the length of $\hat{C}_i$ matters, which we denote as w_i. Because $\hat{C}$ has total length 1, w_i also represents the fraction of the cake $\hat{C}$ allocated to agent i. Next, we view w_i as the *entitlement* (or *weight*) of agent i to the real cake C, and obtain the actual allocation of cake C via a procedure known as the *weighted proportional allocation*.

Weighted proportional cake-cutting. This concept generalizes the proportional cake-cutting to the weighted case. Formally, assume that every agent $i \in N$ is assigned a non-negative *weight* w_i, such that $\sum_{i \in N} w_i = 1$. We call the vector of weights $\mathbf{w} = (w_1, w_2, \ldots, w_n)$ a *weight profile*.

Definition 4.5 (WPR)**.** Given a weight profile $\mathbf{w}$, an allocation $\mathcal{C} = (C_1, C_2, \ldots, C_n)$ of cake C is said to satisfy *weighted proportionality (WPR)* if for every agent $i \in N$, $u_i(C_i) \geq w_i \cdot u_i(C)$.

A weighted proportional allocation of cake gives each agent at least her entitled fraction of the entire cake from her own perspective. The proportionality fairness (Definition 2.1) is a special case of WPR with weight profile $\mathbf{w} = (1/n, 1/n, \ldots, 1/n)$. With any set of agents and any weight profile, a weighted proportional allocation always exists (Cseh and Fleiner 2018). In the following, we will assume that our algorithm is equipped with a protocol WPRALLOC$(N, C, \mathbf{w})$ that could return us a weighted proportional allocation of cake C, among the set of agent N with weight profile $\mathbf{w}$.

The complete algorithm to compute an α-MMS allocation of mixed goods for any number of agents is shown in Algorithm 2. To show that this algorithm can find an α-MMS allocation with mixed goods that contain a heterogeneous cake, it suffices to prove the following two simple facts.

1. $\mathrm{MMS}_i(n, M \cup C) = \mathrm{MMS}_i(n, M \cup \hat{C})$. This is obvious because both C and $\hat{C}$ are divisible with $u_i(C) = u_i(\hat{C})$. Only changing the density of a cake will not affect the MMS value of any agent.
2. $u_i(C_i) \geq u_i(\hat{C}_i)$. This is because by weighted proportionality, we have
$$u_i(C_i) \geq w_i \cdot u_i(C) = w_i \cdot u_i(\hat{C}) = u_i(\hat{C}_i).$$

4.3 Computation

We investigate the computational issues in finding an α-MMS allocation in this part. Note that Algorithm 2 is not a polynomial time algorithm unless P=NP. This is because it requires the knowledge of every agent's MMS value, which is NP-hard to compute even with only indivisible resources (Kurokawa, Procaccia, and Wang 2018).

To obtain a polynomial time approximation algorithm, we first show how to approximate the MMS value of an agent with mixed goods, then focus on obtaining an approximate α-MMS allocation.

Approximate MMS value with mixed goods. When goods are indivisible, Woeginger (1997) showed a polynomial-time approximation scheme (PTAS) to approximately compute the MMS value of an agent. More specifically, given any constant $\epsilon > 0$ and any agent, we can partition the indivisible goods into n bundles in polynomial time, such that each bundle is worth at least $1 - \epsilon$ of that agent's MMS value. By utilizing this PTAS from Woeginger (1997), here we present a new PTAS to approximate MMS values for mixed goods.

Lemma 4.6. *Given any mixed goods instance $I = \langle N, M \cup C \rangle$ and constant $\epsilon > 0$, for any agent $i \in N$, one can compute a partition $(P_1, P_2, \ldots, P_n)$ of $M \cup C$ in polynomial time, such that $\min_{j \in N} u_i(P_j) \geq (1 - \epsilon) \cdot \mathit{MMS}_i(n, M \cup C)$.*

Lemma 4.6 also implies that in the mixed goods setting, we can compute in polynomial time a value MMS'_i such that $\mathrm{MMS}_i \geq \mathrm{MMS}'_i \geq (1 - \epsilon)\mathrm{MMS}_i$.

Approximate α-MMS allocation. Now we turn to the polynomial-time algorithm for computing an approximate α-MMS allocation.

The algorithm is almost similar to Algorithm 2 except for

1. at line 1 of Algorithm 1, we compute the approximate values MMS'_i, which is at most MMS_i and at least $(1 - \epsilon) \cdot \mathrm{MMS}_i$ for each agent $i \in N$;
2. at line 2 of Algorithm 1, we compute the ratio α' using the approximate values MMS', i.e., $\alpha' \leftarrow \min\left\{1, \frac{1}{2} + \min_{i \in N}\left\{\frac{u_i(C)}{2(n-1) \cdot \mathrm{MMS}'_i}\right\}\right\}$.

A similar analysis to Lemma 4.4 shows that the new algorithm with these approximate values will still terminate.

According to Lemma 4.6, we know $\mathrm{MMS}_i \geq \mathrm{MMS}'_i$ for each $i \in N$, which implies that $\alpha' \geq \alpha$. Next, for any agent i, by the design of the algorithm, she is guaranteed a bundle with value at least $\alpha' \cdot \mathrm{MMS}'_i \geq (1 - \epsilon)\alpha' \cdot \mathrm{MMS}_i$. Therefore the resulting allocation is $(1 - \epsilon)\alpha'$-MMS.

Time complexity analysis. In light of Lemma 4.6, computing approximate MMS values takes polynomial time. Then the only step that needs time complexity analysis is the weighted proportional allocation protocol WPRALLOC$(N, C, \mathbf{w})$ at line 4 of Algorithm 2. When all weights are rational numbers, Cseh and Fleiner (2018) gave an implementation of the protocol using $O(n \log D)$ queries, where D is the common denominator of weights. They also showed that their implementation is asymptotically the fastest possible.

We have assumed that our input has size at most L bits. Then each of the arithmetic operations in steps before line 4 (Algorithm 2) keeps the numbers rational with polynomial bit size. Thus, by applying the protocol from Cseh and Fleiner (2018), WPRALLOC at line 4 of Algorithm 2 can be implemented in polynomial time. Summarize everything together, we obtain a polynomial-time algorithm.

Sections 4.1, 4.2 and 4.3 together complete the proof of Theorem 4.1.

4.4 Boosting the Approximation Ratio

In Theorem 4.1, the smallest value for α is $\frac{1}{2}$, achieved when the resources contain only indivisible goods. In this case, the theorem ensures that a $\frac{1}{2}$-MMS allocation always exists. However, there is a gap between this $\frac{1}{2}$ guarantee from our result and that of the currently best-known result with only indivisible goods, which is $\gamma_I \geq \frac{3}{4} + \frac{1}{12n}$ according to Garg and Taki (2020). In the following, we show that a simple procedure can boost the MMS approximation ratio computed by our algorithm to (almost) match the currently best-known ratio for indivisible goods.

First, existence-wise, combining Theorem 4.1 with Corollary 3.3 ($\gamma_I = \gamma_M$), we can improve ratio directly to $\max\{\alpha, \gamma_I\}$ in Theorem 4.1. Next, computation-wise, suppose there exists a polynomial-time algorithm that guarantees to output a β-MMS allocation with indivisible goods for some β. Then given a mixed good problem instance, we first compute α' via Theorem 4.1 and compare it with β: if $\alpha' \geq \beta$, we directly apply Theorem 4.1; otherwise, we cut the cake C into small intervals, each valued at most $\frac{\epsilon \cdot u_i(C)}{2n}$ for each agent i, and use the β-MMS algorithm to obtain the allocation of this instance with only indivisible goods. In summary, we have the following strengthened result:

Theorem 4.7. *A $\max\{\alpha, \gamma_I\}$-MMS allocation with mixed goods lways exists for any number of agents.*

In addition, if there exists a polynomial-time algorithm that can always output a β-MMS allocation with indivisible goods, then for any constant $\epsilon > 0$, there is another polynomial-time algorithm that computes a $(1 - \epsilon)\max\{\alpha', \beta\}$-MMS allocation with mixed goods.

The proof of Theorem 4.7 utilizes the proof of Lemma 4.6 and is straightforward to prove. The currently best lower bound of γ_I is $\gamma_I \geq \frac{3}{4} + \frac{1}{12n}$ and the currently best-known value of β is $\frac{3}{4}$, both are due to Garg and Taki (2020). Any better lower bound of γ_I and value of β found in the future would immediately imply a better MMS approximation guarantee in the mixed goods setting as well.

5 Relation of MMS and EFM

Proportionality fairness, and its generalization, MMS, are often compared to another well studied fairness notion of *envy-freeness (EF)*. It is known that with only divisible goods, envy-freeness implies proportionality but not vice versa. With only indivisible goods, the relaxed notion of EF, known as *envy-freeness up to one item (EF1)*, and the relaxed notion of proportionality, MMS, do not imply each other (Caragiannis et al. 2016). In a recent work, Bei et al. (2020a) proposed a new envy-freeness notion, termed *envy-freeness for mixed goods (EFM)*, that generalizes both EF and EF1 to the mixed goods setting. We include the definition of EFM as follows.

Definition 5.1 (EFM)**.** An allocation $\mathcal{A}$ is said to satisfy *envy-freeness for mixed goods (EFM)* in the sense that for any $i, j \in N$,

- if j's bundle consists of only indivisible goods, there exists $g \in A_j$ such that $u_i(A_i) \geq u_i(A_j \setminus \{g\})$;
- otherwise, $u_i(A_i) \geq u_i(A_j)$.

As, with only indivisible goods, EFM reduces to EF1, it is obvious to see that neither EFM nor MMS implies the other. We then consider the relation between EFM and the approximation of MMS, focusing on what approximation ratio of MMS can be achieved by an EFM allocation.

On the one hand, when all goods are divisible, EFM (or EF) is always 1-MMS (or proportionality). On the other hand, when all goods are indivisible, Amanatidis, Birmpas, and Markakis (2018) showed that any EFM (or EF1) allocation is always $\frac{1}{n}$-MMS and this approximation ratio is tight. Then, with mixed goods, one might ask if an EFM allocation would have the MMS approximation ratio laying between $\frac{1}{n}$ and 1. Our next lemma confirms this conjecture.

Lemma 5.2. *Given any mixed goods instance $(N, M \cup C)$, for any EFM allocation $(A_1, A_2, \ldots, A_n)$ and any agent $i \in N$, we have*

$$v_i(A_i) \geq \frac{\mathit{MMS}_i(n, M) + v_i(C)}{n} \geq \frac{\mathit{MMS}_i(n, M \cup C)}{n}.$$

The proof is a direct generalization of the proof of Proposition 3.6 in (Amanatidis, Birmpas, and Markakis 2018).

From Lemma 5.2, we know that EFM implies α-MMS where α is a monotonically increasing function that depends on the agent's value on the whole cake. In other words, one can directly utilize the EFM allocation to obtain an α-MMS allocation with α varied from $1/n$ (when goods are indivisible only) to 1 (when goods are divisible only). On the other hand, our result in Section 4 shows that we can always have an α-MMS allocation with α ranging from $1/2$ to 1.

6 Conclusion and Future Work

In this paper, we have studied the extent to which we can find approximate MMS allocations when the resources contain both divisible and indivisible goods. We analyzed the relation of the worst-case MMS approximation guarantees between mixed goods instances and indivisible goods instances. We also presented an algorithm to produce an α-MMS allocation for any number of agents, where α monotonically increases in terms of the ratio between agents' values for the divisible goods and their MMS values. For future work, it would be interesting to improve the MMS approximation guarantee with mixed goods. Another working direction is to study fair allocations in the mixed goods setting in conjunction with economic efficiency notions such as Pareto optimality.

Acknowledgments

This project is supported by the Ministry of Education, Singapore, under its Academic Research Fund Tier 1 (RG23/20).

References

Abdulkadiroğlu, A.; Sönmez, T.; and Ünver, M. U. 2004. Room Assignment-Rent Division: A Market Approach. *Social Choice and Welfare* 22(3): 515–538.

Alkan, A.; Demange, G.; and Gale, D. 1991. Fair Allocation of Indivisible Goods and Criteria of Justice. *Econometrica* 59(4): 1023–1039.

Amanatidis, G.; Birmpas, G.; and Markakis, E. 2018. Comparing Approximate Relaxations of Envy-Freeness. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence (IJCAI)*, 42–48.

Amanatidis, G.; Markakis, E.; Nikzad, A.; and Saberi, A. 2017. Approximation Algorithms for Computing Maximin Share Allocations. *ACM Transactions on Algorithms* 13(4): 52:1–52:28.

Arunachaleswaran, E. R.; Barman, S.; and Rathi, N. 2019. Fully Polynomial-Time Approximation Schemes for Fair Rent Division. In *Proceedings of the 30th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 1994–2013.

Aziz, H. 2021. Achieving Envy-freeness and Equitability with Monetary Transfers. In *Proceedings of the 35th AAAI Conference on Artificial Intelligence (AAAI)*. Forthcoming.

Barman, S.; and Krishnamurthy, S. K. 2020. Approximation Algorithms for Maximin Fair Division. *ACM Transactions on Economics and Computation (TEAC)* 8(1): 5:1–5:28.

Bei, X.; Li, Z.; Liu, J.; Liu, S.; and Lu, X. 2020a. Fair Division of Mixed Divisible and Indivisible Goods. In *Proceedings of the 34th AAAI Conference on Artificial Intelligence (AAAI)*, 1814–1821.

Bei, X.; Liu, S.; Lu, X.; and Wang, H. 2020b. Maximin Fairness with Mixed Divisible and Indivisible Goods. *CoRR* abs/2002.05245. URL https://arxiv.org/abs/2002.05245.

Bhaskar, U.; Sricharan, A.; and Vaish, R. 2020. On Approximate Envy-Freeness for Indivisible Chores and Mixed Resources. *CoRR* abs/2012.06788. URL https://arxiv.org/abs/2012.06788.

Bouveret, S.; Cechlárová, K.; Elkind, E.; Igarashi, A.; and Peters, D. 2017. Fair Division of a Graph. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence (IJCAI)*, 135–141.

Brams, S. J. 2008. *Mathematics and Democracy: Designing Better Voting and Fair-Division Procedures*. Princeton University Press.

Brams, S. J.; and Taylor, A. D. 1996. *Fair Division: From Cake-Cutting to Dispute Resolution*. Cambridge University Press.

Brustle, J.; Dippel, J.; Narayan, V.; Suzuki, M.; and Vetta, A. 2020. One Dollar Each Eliminates Envy. In *Proceedings of the 21st ACM Conference on Economics and Computation (EC)*, 23–39.

Budish, E. 2011. The Combinatorial Assignment Problem: Approximate Competitive Equilibrium from Equal Incomes. *Journal of Political Economy* 119(6): 1061–1103.

Caragiannis, I.; and Ioannidis, S. 2020. Computing Envy-freeable Allocations with Limited Subsidies. *CoRR* abs/2002.02789. URL https://arxiv.org/abs/2002.02789.

Caragiannis, I.; Kurokawa, D.; Moulin, H.; Procaccia, A. D.; Shah, N.; and Wang, J. 2016. The Unreasonable Fairness of Maximum Nash Welfare. In *Proceedings of the 17th ACM Conference on Economics and Computation (EC)*, 305–322.

Cseh, Á.; and Fleiner, T. 2018. The Complexity of Cake Cutting with Unequal Shares. In *Proceedings of the 11th International Symposium on Algorithmic Game Theory (SAGT)*, 19–30.

Farhadi, A.; Ghodsi, M.; Hajiaghayi, M.; Lahaie, S.; Pennock, D.; Seddighin, M.; Seddighin, S.; and Yami, H. 2019. Fair Allocation of Indivisible Goods to Asymmetric Agents. *Journal of Artificial Intelligence Research (JAIR)* 64(1): 1–20.

Gal, Y. K.; Mash, M.; Procaccia, A. D.; and Zick, Y. 2017. Which Is the Fairest (Rent Division) of Them All? *Journal of the ACM (JACM)* 64(6): 39:1–39:22.

Garg, J.; McGlaughlin, P.; and Taki, S. 2019. Approximating Maximin Share Allocations. In *Proceedings of the 2nd Symposium on Simplicity in Algorithms (SOSA)*, 20:1–20:11.

Garg, J.; and Taki, S. 2020. An Improved Approximation Algorithm for Maximin Shares. In *Proceedings of the 21st ACM Conference on Economics and Computation (EC)*, 379–380.

Ghodsi, M.; Hajiaghayi, M.; Seddighin, M.; Seddighin, S.; and Yami, H. 2018. Fair Allocation of Indivisible Goods: Improvements and Generalizations. In *Proceedings of the 19th ACM Conference on Economics and Computation (EC)*, 539–556.

Goldman, J.; and Procaccia, A. D. 2015. Spliddit: Unleashing Fair Division Algorithms. *SIGecom Exchanges* 13(2): 41–46.

Gourvès, L.; and Monnot, J. 2019. On Maximin Share Allocations in Matroids. *Theoretical Computer Science* 754: 50–64.

Haake, C.-J.; Raith, M. G.; and Su, F. E. 2002. Bidding for Envy-Freeness: A Procedural Approach to n-Player Fair-Division Problems. *Social Choice and Welfare* 19(4): 723–749.

Halpern, D.; and Shah, N. 2019. Fair Division with Subsidy. In *Proceedings of the 12th International Symposium on Algorithmic Game Theory (SAGT)*, 374–389.

Igarashi, A.; and Peters, D. 2019. Pareto-Optimal Allocation of Indivisible Goods with Connectivity Constraints. In *Proceedings of the 33rd AAAI Conference on Artificial Intelligence (AAAI)*, 2045–2052.

Klijn, F. 2000. An Algorithm for Envy-free Allocations in an Economy with Indivisible Objects and Money. *Social Choice and Welfare* 17(2): 201–215.

Kurokawa, D.; Procaccia, A. D.; and Wang, J. 2018. Fair Enough: Guaranteeing Approximate Maximin Shares. *Journal of the ACM (JACM)* 65(2): 8:1–8:27.

Lonc, Z.; and Truszczynski, M. 2020. Maximin Share Allocations on Cycles. *Journal of Artificial Intelligence Research (JAIR)* 69: 613–655.

Maskin, E. S. 1987. On the Fair Allocation of Indivisible Goods. In Feiwel, G. R., ed., *Arrow and the Foundations of the Theory of Economic Policy*, chapter 11, 341–349. Palgrave Macmillan UK.

Meertens, M.; Potters, J.; and Reijnierse, H. 2002. Envy-free and Pareto Efficient Allocations in Economies with Indivisible Goods and Money. *Mathematical Social Sciences* 44(3): 223–233.

Steinhaus, H. 1948. The Problem of Fair Division. *Econometrica* 16(1): 101–104.

Su, F. E. 1999. Rental Harmony: Sperner's Lemma in Fair Division. *The American Mathematical Monthly* 106(10): 930–942.

Suksompong, W. 2018. Approximate Maximin Shares for Groups of Agents. *Mathematical Social Sciences* 92: 40–47.

Woeginger, G. J. 1997. A Polynomial-Time Approximation Scheme for Maximizing the Minimum Machine Completion Time. *Operations Research Letters* 20(4): 149 – 154.

Young, H. P. 1995. *Equity: In Theory and Practice*. Princeton University Press.