Autonomous Agents and Multi-Agent Systems (2021) 35:34
https://doi.org/10.1007/s10458-021-09517-7

Maximin fairness with mixed divisible and indivisible goods

Xiaohui Bei[1] · Shengxin Liu[2] · Xinhang Lu[1] · Hongao Wang[1]

Accepted: 12 June 2021 / Published online: 30 June 2021

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.

Keywords Fair division · Mixed goods · Maximin fairness

A preliminary version appeared in Proceedings of the 35th AAAI Conference on Artificial Intelligence (AAAI) [17]. This project/research is supported by the Ministry of Education, Singapore, under its Academic Research Fund Tier 2 (MOE2019-T2-1-045).

✉ Shengxin Liu
sxliu@hit.edu.cn

Xiaohui Bei
xhbei@ntu.edu.sg

Xinhang Lu
xinhang001@e.ntu.edu.sg

Hongao Wang
hongao.wang@ntu.edu.sg

[1] School of Physical and Mathematical Sciences, Nanyang Technological University, Jurong West, Singapore

[2] School of Computer Science and Technology, Harbin Institute of Technology, Shenzhen, China

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, e.g., [23, 24, 33, 53, 59] for a survey). In particular, as one of the most classic and widely known fairness notions, Steinhaus [55] 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 [55] also showed that a proportional allocation can always be found for any number of agents over any divisible good.[1] 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 [26] 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 that 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 et al. [48] showed that even with additive valuation functions, an MMS allocation may not always exist when there are at least three agents.[2] However, Kurokawa et al. showed that 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 [48] 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 (for which the current best ratio is $3/4 + 1/(12n)$ due to Garg and Taki [38]), design of simpler algorithms, etc. [4, 14, 39, 40]. MMS has also been adopted as the fairness solution concept in several practical applications [26, 41].

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. [16], in which the authors proposed a new fairness notion called *envy-freeness for mixed goods*

[1] Such a solution was credited to B. Knaster and S. Banach by Steinhaus [55].

[2] An MMS allocation always exists when there are two agents [20].

(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 best possible 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 Sect. 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 into 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 the cake.

Next in Sect. 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 $(1-\epsilon)$ α-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, and L is the input bit length.

Last, in Sect. 5, we discuss the relation between MMS and the recently introduced notion of envy-freeness for mixed goods (EFM) in the mixed goods setting. Generally speaking, neither MMS nor 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

As mentioned earlier, proportionality fairness was first introduced seven decades ago in the seminal work by Steinhaus [55] in the context of cake-cutting. Since then, several efficient algorithms have been proposed [31, 34], and a matching lower bound has been found by Edmonds and Pruhs [32].

Despite the fully fledged theory of cake cutting [13, 23, 54], it was not until recently attracted significant attention on fair division of indivisible goods. Maximin share (MMS) fairness, often regarded as a generalization of proportionality to indivisible resource allocation, was first introduced by Budish [26]. Kurokawa et al. [48] later showed that an MMS allocation may not always exist, but a 2/3-MMS allocation always exists for any number of agents. Amanatidis et al. [4] then devised an algorithm that computes a $(2/3-\epsilon)$-MMS allocation with running time polynomial in the number of agents and goods. The approximation guarantee for MMS was further improved to 3/4 by Ghodsi et al. [40] and the currently best-known ratio is $3/4 + 1/(12n)$ due to Garg and Taki [38].

In addition to the works we mentioned above, MMS allocations of indivisible resources have also been extensively studied in several other settings, including for agents with unequal entitlements [35] or in different groups [57], for goods forming an undirected graph [21, 50], for allocations under matroid constraints [42] or in conjunction with economic efficiency [46], as well as in the context of *chore division*, where chores refer to negatively valued items [9, 11, 12, 14, 45]. Caragiannis et al. [28] introduced *pairwise maximin share guarantee*, which is incomparable with MMS fairness. It remains an open problem question whether a pairwise MMS allocation always exists [28]. Barman et al.[15] defined a stronger fairness notion than MMS, called *groupwise maximin share guarantee (GMMS)*, and showed that GMMS allocations always exist in specific settings.

Besides proportionality and MMS fairness, another prominent fairness notion in resource allocation is *envy-freeness (EF)*, which requires that each agent weakly prefers her own bundle to any other agent's bundle [36]. It follows from definition that envy-freeness implies proportionality when agents have additive valuations and all goods must be allocated. An envy-free allocation of divisible goods for any number of agents always exists [3] and can be computed via a discrete and bounded algorithm [8]. With indivisible goods, envy-freeness may not always be achievable. This notion is then often relaxed to *envy-freeness up to one good (EF1)*, which requires that any envy an agent has towards another agent can be eliminated by removing some good from the latter's bundle. An EF1 allocation always exists for any number of agents and can be computed efficiently [49].

Recently, Bei et al.[16] initiated the study of fair allocation of mixed divisible and indivisible goods. They introduced the fairness notion of *envy-freeness for mixed goods (EFM)*, which unifies EF and EF1 in the mixed good setting. Bei et al. showed that an EFM allocation with mixed goods always exists for any number of agents and also investigated its computational aspects. Later, Bhaskar et al. [18] established an analogous existence result in a setting where *undesirable* indivisible items and a cake are to be divided as well as a similar result in the flipped setting consisting of indivisible goods and *undesirable* divisible items.

A related line of research incorporates money into the fair division of indivisible goods, with the consideration of finding envy-free allocations [2, 7, 25, 27, 44, 47, 51, 52]. In a recent work, Halpern and Shah [44] bounded the amount of money needed to achieve envy-freeness for agents with additive valuations, assuming that the value of each agent for each good is at most 1. Their results were further improved by Brustle et al. [25]. On the other hand, Caragiannis and Ioannidis [27] studied the optimization problem of computing the minimum amount of subsidy needed to obtain envy-freeness given an allocation instance and showed both hardness and approximation results. Aziz [7] studied the problem of using monetary transfers to achieve envy-freeness and equitability simultaneously.[3] Our work and previous works indeed share lots of similarities, because money is just a special type of cake. However, a major difference between our work and theirs is that we have different objectives. This also makes our results and theirs incomparable. More specifically, their works aim to *determine* the amount of money needed to be added to the set of indivisible goods such that an envy-free allocation can be obtained. However, their results do not specify what to do when there is not enough money as they required. In our work, we focus on instances in which the amount of indivisible and divisible goods (or cake) are both *fixed*, and regardless of whether the cake is enough to guarantee an MMS allocation, we aim to find a reasonably fair allocation with an MMS approximation guarantee.

Another closely related problem is *rent division* (see, e.g., [1, 6, 22, 37, 43, 56]). 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 $C = [0, 1]$. 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. 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 of cake $S \subseteq [0, 1]$, agent i's value over S is then defined as $u_i(S) := \int_{x \in S} f_i(x)\, dx$. 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. Let $\mathcal{M} = (M_1, M_2, \dots, M_n)$ be a partition of indivisible goods M into n bundles such that agent i receives M_i. Let $\mathcal{C} = (C_1, C_2, \dots, C_n)$ be a partition of the cake C such that agent i gets a piece of cake C_i. An *allocation* of mixed

[3] An allocation is said to satisfy *equitability* if every agent gets the same utility.

goods $G = M \cup C$ is defined as $\mathcal{A} = (A_1, A_2, \dots, A_n)$, 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 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 *** Let $\Pi_k(G) = \{\{P_1, P_2, \dots, P_k\} \mid P_i \cap P_j = \emptyset \, \forall i \neq j \text{ and } \bigcup_k P_k = G\}$ be the set of k-partitions of G. Define the k-maximin share of agent i as

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

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

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

Definition 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.[4] 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 the leftmost point y such that $u_i([x, y]) = \beta$.

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

[4] While an MMS allocation may not exist in general, such an allocation always exists in the case of two agents.

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 [48]. 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 4 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.[5] 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 shown that $\gamma_I < 1$ [48] and $\gamma_I \geq \frac{3}{4} + \frac{1}{12n}$ [38].

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 1 *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 into 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 $i \in N$.*

Proof We first transform the mixed goods instance $I = \langle N, M \cup C\rangle$ into an instance $I' = \langle N, M'\rangle$ with only indivisible goods. Consider an agent i and an MMS partition $\mathcal{P}_i$ for this agent in I. Assume that $\mathcal{P}_i$ divides cake C into at most n intervals with at most $n-1$ cuts. This assumption is without loss of generality because in an MMS partition it only matters how much value worth of cake is assigned to each bundle, but not their positions. Then, by collecting all cuts of all n MMS partitions $\mathcal{P}_1, \ldots, \mathcal{P}_n$ on C, they cut the cake into at most $n(n-1)+1$ pieces. We can treat these pieces on C as a set M'' of indivisible "frozen pieces". Together with M, we now have $M' = M'' \cup M$.

[5] 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.

Given any allocation $\mathcal{A}'$ of M', we can easily convert it into an allocation $\mathcal{A}$ of G by transforming those "frozen cake pieces" back to normal cake pieces. This also gives $u_i(A_i) = u_i(A'_i)$ for each agent $i \in N$, which proves the first part of Theorem 1.

Last, it is clear that every agent can have the same MMS partition in I' as that in I, because the cuts do not affect their MMS partitions. This implies that $\mathrm{MMS}_i(n, M') \geq \mathrm{MMS}_i(n, M \cup C)$ for each agent $i \in N$. On the other hand, the first part of this theorem also implies $\mathrm{MMS}_i(n, M \cup C) \geq \mathrm{MMS}_i(n, M')$. Hence we have $\mathrm{MMS}_i(n, M \cup C) = \mathrm{MMS}_i(n, M')$ 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 1 directly implies the following result.

Corollary 1 $\gamma_I = \gamma_M$.

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 Sect. 4.

3.2 Cake does not always help

Note that the equation in Corollary 1 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 $\gamma(I)$.[6]

Theorem 2 *For any $n \geq 6$, 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.

Before showing the detailed proof, we first 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 $\mathrm{MMS}_i(n, M \cup C)$ should be exactly $\mathrm{MMS}_i(n, M) + \epsilon$. Now suppose that in the instance I, all of 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

[6] In the indivisible setting, the corresponding result of adding an item may lower the best MMS guarantee for a problem instance is easy to get.

$\gamma(\langle N, M\rangle)$-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 of 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 [48] can be utilized to construct the instance I that satisfies all above mentioned properties. By utilizing their construction, our argument requires at least six agents. The full proof can be found in the following.

Proof Our counterexample will utilize the following lemma from [48], which is also used for showing the non-existence of full MMS allocation with indivisible goods.

Lemma 1 (Base of counterexample [48]) *For any* $n \geq 6$, *there exists an* $n \times n$ *matrix* M, *satisfying the following properties*:

1. *All entries are non-negative* ($i.e.$, $\forall i, j : M_{i,j} \geq 0$).
2. *All entries of the last row and column, and the first entry in the first row, are positive* ($i.e.$, $\forall i : M_{i,n}, M_{n,i} > 0$ *and* $M_{1,1} > 0$).
3. *All rows and columns sum to* 1 ($i.e.$, $M\mathbf{1} = M^{\top}\mathbf{1} = \mathbf{1}$).
4. *Define* M^{+} *as the set of all positive entries in* M. *Then if we wish to partition* M^{+} *into* n *subsets that sum to exactly* 1, *then our partition must correspond to either the rows of* M *or the columns of* M.

Then construct two $n \times n$ matrices P^{+} and P^{-}. Let $P^{+}_{1,1} = P^{-}_{1,1} = -\epsilon$, $P^{+}_{n,1} = P^{-}_{1,n} = -\epsilon$, $P^{+}_{n,n} = P^{-}_{n,n} = (2n-3)\epsilon$, and $P^{+}_{n,i} = P^{-}_{i,n} = -2\epsilon$ for $2 \leq i \leq n-1$. Take $n = 6$ as an example, we show below the construction of matrices P^{+} and P^{-}:

$$P^{+} = \begin{bmatrix} -\epsilon & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ -\epsilon & -2\epsilon & -2\epsilon & -2\epsilon & -2\epsilon & 9\epsilon \end{bmatrix}$$

$$P^{-} = \begin{bmatrix} -\epsilon & 0\ 0\ 0\ 0 & -\epsilon \\ 0 & 0\ 0\ 0\ 0 & -2\epsilon \\ 0 & 0\ 0\ 0\ 0 & -2\epsilon \\ 0 & 0\ 0\ 0\ 0 & -2\epsilon \\ 0 & 0\ 0\ 0\ 0 & -2\epsilon \\ 0 & 0\ 0\ 0\ 0 & 9\epsilon \end{bmatrix}$$

Consider a matrix M satisfying all properties listed in Lemma 1. By setting a properly small value ϵ, we can always make sure that every entry of $M + P^{+}$ and $M + P^{-}$ is non-negative. In the following, we will treat each entry as an indivisible good. We next divide N into two disjoint subsets. One contains $\left\lfloor \frac{n}{2} \right\rfloor$ agents, denoted by N^{+}. The other contains the rest of the agents, denoted by N^{-}. We let each agent $i \in N^{+}$ take the values of n^2 items as in matrix $M + P^{+}$, and each agent $i \in N^{-}$ take the values of n^2 items as in matrix $M + P^{-}$. We call this problem instance I. One can check that in this instance I, the MMS value for each agent i is $1 - \epsilon$.

According to the fourth property in Lemma 1, there are only two ways to distribute these items into n bundles such that each bundle has value close to 1: either the rows of M or the columns of M. It means that once fixing the partition (by rows or by columns), half of the agents may receive bundles with value $1 - 2\epsilon$ or the bundle with value more than 1. In each of these two partitions, due to $n \geq 6$, we can always find at least 2 agents who value their bundles exactly $1 - 2\epsilon$. For example, there are two such agents from N^+ if the partition is n columns, or two from N^- if the partition is n rows. In particular, this means the MMS approximation guarantee $\gamma(I)$ of this instance I is $\frac{1-2\epsilon}{1-\epsilon}$.

Suppose now we add a homogeneous cake to this problem instance I. This cake has value ϵ for each agent. Every agent's MMS value will now increase from $1 - \epsilon$ to 1. However, in any allocation, there will still be at least two agents whose values for the indivisible goods are no more than $1 - 2\epsilon$. Then the best possible way to distribute the cake is to allocate it only to those agents, which means at least one such agent will receive a bundle of value at most $1 - 2\epsilon + \frac{\epsilon}{2} = 1 - \frac{3\epsilon}{2}$. Thus, in this case, the MMS approximation ratio of such agent will be no more than $1 - \frac{3\epsilon}{2}$, which is strictly smaller than $\frac{1-2\epsilon}{1-\epsilon}$ when $\epsilon < 1/4$. □

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 mixed goods problem instances. 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 3 *Given any mixed good problem instance $\langle N, M \cup C \rangle$, an α-MMS allocation always exists, where*

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

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 3 has several implications. For example, when every agent i has $u_i(C) \geq (n/2)\text{MMS}_i$, Theorem 3 implies the existence of an α-MMS allocation with α better than the currently best-known approximation ratio of $\frac{3}{4} + \frac{1}{12n}$ with indivisible goods due to Garg and Taki [38]. In addition, the following corollary shows the amount of divisible goods needed to ensure that the instance admits a full-MMS allocation.

Corollary 2 *Given a mixed good problem instance* $I = \langle N, M \cup C \rangle$, *if* $u_i(C) \geq (n-1)\text{MMS}_i$ *holds for each agent* $i \in N$, *then an MMS allocation is guaranteed to exist.*

This means that even with the presence of indivisible items, as long as there are enough divisible goods, a full-MMS allocation can always be found. However, we note that this corollary should not be interpreted as that this is the *least* amount of divisible goods required. For example, Halpern and Shah [44] and Brustle et al. [25] studied the allocation of indivisible goods and a very special type of divisible goods, money. They bounded the amount of money needed for an indivisible goods instance to have an envy-free allocation, assuming that the value of each agent for each good is at most 1. Although an envy-free allocation is also a full-MMS allocation, their results and this corollary are incomparable because we have different objectives, and it is not our goal to find the minimum amount of cake needed to ensure an MMS allocation.

The remaining of this section is dedicated to the proof of Theorem 3. 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 into 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 Sect. 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. 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 $u_i(S) := (\sum_{[a,b]\in S}(b-a))u_i(\hat{C})$.

4.1.1 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 [40]. 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 a 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 13). Then again, both the agent and her goods are removed from the instance.

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 $\mathrm{MMS}_i$, for each $i \in N$.
 2 $\alpha \leftarrow \min\left\{1, \frac{1}{2} + \min_{i \in N}\left\{\frac{u_i(\hat{C})}{2(n-1)\cdot \mathrm{MMS}_i}\right\}\right\}$
 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 $u_j(B) \geq (1-\alpha)\cdot \mathrm{MMS}_j$ 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
         $u_i(B \cup [a, x_i]) \geq \alpha \cdot \mathrm{MMS}_i$.
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 $(A_1, A_2, \ldots, A_n)$
```

4.1.2 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, it follows from Lemma 1 of Bouveret and Lemaître [20] 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 2 (Monotonicity property)[7] *Given an instance $\langle N, G = M \cup C \rangle$, for any agent $i \in N$ and any indivisible good $g \in M$, it holds that*

[7] We adopt the name "monotonicity property" from Amanatidis et al. [4].

$$\mathrm{MMS}_i(n-1, G \setminus \{g\}) \geq \mathrm{MMS}_i(n, G).$$

Proof Removing a single indivisible good in an MMS partition of agent i affects exactly one bundle and each of the remaining $n-1$ bundles has value at least $\mathrm{MMS}_i(n, G)$. Therefore, we have $\mathrm{MMS}_i(n-1, G \setminus \{g\}) \geq \mathrm{MMS}_i(n, G)$. □

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 2) $n - n_1$ times, we have that for each agent $i \in N_1$, $\mathrm{MMS}_i(n_1, G_1) \geq \mathrm{MMS}_i(n, G)$. In addition, each agent i who leaves the system in this phase receives an item of value at least $\alpha \cdot \mathrm{MMS}_i$. 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 $\alpha \cdot \mathrm{MMS}_i$.

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) $u_{i^*}(A_{i^*}) \geq \alpha \cdot \mathrm{MMS}_{i^*}$;
(2) For each agent j remaining in N, $u_j(A_{i^*}) \leq \mathrm{MMS}_j$.

(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 $\alpha \cdot \mathrm{MMS}_i$ for any agent i. Therefore, the set B selected at line 9 must satisfy $u_j(B) \leq \mathrm{MMS}_j$ for all $j \in N$. In line 10, each agent cuts a piece of cake such that the sum of her value for B and this piece of cake is at least α fraction of her maximin share. Because $\alpha \leq 1$ and the cake is divisible, after line 10, it continues to satisfy that $u_j(B \cup [a, x_j]) \leq \mathrm{MMS}_j$ for each $j \in N$. Then, because i^* is selected such that x_{i^*} is the smallest value, one would have $u_j(A_{i^*} = B \cup [a, x_{i^*}]) \leq u_j(B \cup [a, x_j]) \leq \mathrm{MMS}_j$ 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 $\alpha \cdot \mathrm{MMS}_i$ after the two phases. It only remains to show that the cake $\hat{C}$ is enough to be allocated throughout the process.

Lemma 3 *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.*

Proof Line 2 in Algorithm 1 indicates that for each agent $i \in N$, $u_i(\hat{C}) \geq (n-1) \cdot (2\alpha - 1) \cdot \mathrm{MMS}_i$. As a result, each agent i has value at least $(2\alpha - 1) \cdot \mathrm{MMS}_i$ for a $\frac{1}{n-1}$ fraction of the entire cake $\hat{C}$. It is also clear that Phase 2 has been executed at most $n-1$ times during the algorithm run. That is to say the action of cutting a piece of $\hat{C}$ and allocating this piece to an agent is performed at most $n-1$ times.

Based on whether there exists some agent who has value at least $1-\alpha$ times her MMS for goods in B (line 9), we distinguish two cases.

- *Line* 9: *there exists some agent j with $u_j(B) \geq (1-\alpha) \cdot \mathrm{MMS}_j$*. As mentioned earlier, a $\frac{1}{n-1}$ fraction of $\hat{C}$ is worth at least $(2\alpha - 1) \cdot \mathrm{MMS}_j$. Thus it together with B is enough to give agent j a value of at least $\alpha \cdot \mathrm{MMS}_j$. This means at line 10, the length of $[a, x_j]$ is no more than $\frac{1}{n-1}$. Moreover, Algorithm 1 chooses the agent who claims the smallest piece of cake as agent i^* at line 11, which means the length of $[a, x_{i^*}]$ is again no more

than $\frac{1}{n-1}$. Combining the fact that Phase 2 executes at most $n-1$ times, if this case holds every time, the cake will be enough.
- *Line* 9: $u_j(B) < (1-\alpha) \cdot \mathrm{MMS}_j$ *for each agent* j. In this case, B is set to be M at line 9. Note that after the first time of such case, M will become empty, and the agents left will divide only the cake for the remaining rounds. Let k be the number of the remaining agents when M becomes empty. By property (2) that we showed above, we know the remaining cake is valued at least $k \cdot \mathrm{MMS}_i$ for each remaining agent i. Thus it is enough for each agent i to receive a piece with value at least $\alpha \cdot \mathrm{MMS}_i$.

The lemma thus follows. □

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 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. Since $\hat{C}$ 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 $\mathbf{w} = (w_1, w_2, \dots, w_n)$ a *weight profile*.

Definition 5 (*WPR*) Given a weight profile $\mathbf{w}$, an allocation $\mathcal{C} = (C_1, C_2, \dots, C_n)$ 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 1) is a special case of WPR with weight profile $\mathbf{w} = (1/n, 1/n, \dots, 1/n)$. With any set of agents and any weight profile, a weighted proportional allocation always exists [30]. In the following, we will assume that our algorithm is equipped with a protocol $\textsc{WPRAlloc}(N, C, \mathbf{w})$ that could return us a weighted proportional allocation of cake C, among the set of agent N with weight profile $\mathbf{w}$.

```
Algorithm 2: The Mixed MMS Algorithm
  Input: Agents $N$, indivisible goods $M$ and cake $C$, utility and density functions.
  Let $\hat{C} = [0, 1]$ be a homogeneous cake with $u_i(\hat{C}) = u_i(C)$ for each agent $i \in N$.
  $(M_1 \cup \hat{C}_1, M_2 \cup \hat{C}_2, \dots, M_n \cup \hat{C}_n) \leftarrow$ Mixed-MMS-Homogeneous($\langle N, M \cup \hat{C} \rangle$)
  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.
  $(C_1, C_2, \dots, C_n) \leftarrow$ WPRAlloc($N, C, \mathbf{w} = (w_1, \dots, w_n)$)    // allocate cake $C$
  return $(M_1 \cup C_1, M_2 \cup C_2, \dots, M_n \cup C_n)$
```

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. $\text{MMS}_i(n, M \cup C) = \text{MMS}_i(n, M \cup \hat{C})$. 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 [48].

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.

4.3.1 Approximate MMS value with mixed goods

When goods are indivisible, Woeginger [58] showed a polynomial-time approximation scheme (PTAS) to approximately compute the MMS value of an agent. More specifically, given any constant $\delta > 0$ and any agent, we can partition the indivisible goods into n bundles in polynomial time, such that each bundle is worth at least $1 - \delta$ of that agent's MMS value. By utilizing this PTAS from Woeginger [58], here we present a new PTAS to approximate MMS values for mixed goods.

Lemma 4 *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 $(P_1, P_2, \dots, P_n)$ of $M \cup C$ in polynomial time, such that* $\min_{j \in N} u_i(P_j) \geq (1 - \epsilon) \cdot \text{MMS}_i(n, M \cup C)$.

Proof Let agent i cut the cake C into $\left\lceil \frac{2n}{\epsilon} \right\rceil$ disjoint intervals worth at most $\frac{\epsilon \cdot u_i(C)}{2n}$ each to this agent. Denote by $\tilde{C}$ the collection of these discretized, indivisible intervals. The new discretized instance is then denoted by $I' = \langle N, M \cup \tilde{C} \rangle$. This is a problem instance with only indivisible goods.

We first claim that

$$\mathrm{MMS}_i(n, M \cup C) \geq \mathrm{MMS}_i(n, M \cup \tilde{C}) \geq \left(1 - \frac{\epsilon}{2}\right) \cdot \mathrm{MMS}_i(n, M \cup C).$$

The first inequality holds trivially by definition. We proceed to show the second. Consider an MMS partition $\mathcal{T}$ of I for agent i. We construct a partition $\mathcal{T}'$ of I' as follows. First, let the partition of its original indivisible goods M be exactly the same as that in $\mathcal{T}$. We then distribute the intervals in $\tilde{C}$ into these n bundles. For any bundle whose value is less than $\left(1 - \frac{\epsilon}{2}\right) \cdot \mathrm{MMS}_i(n, M \cup C)$ to agent i, add one interval at a time to this bundle until agent i's value for this bundle falls in $\left[\left(1 - \frac{\epsilon}{2}\right)\mathrm{MMS}_i(n, M \cup C), \mathrm{MMS}_i(n, M \cup C)\right]$. This is possible because $\mathrm{MMS}_i(n, M \cup C) \geq u_i(C)/n$ and each interval is worth at most $\frac{\epsilon \cdot u_i(C)}{2n} \leq \frac{\epsilon}{2} \cdot \mathrm{MMS}_i(n, M \cup C)$. Also $\tilde{C}$ will have enough pieces for these allocations because in $\mathcal{T}$, each bundle is worth at least $\mathrm{MMS}_i(n, M \cup C)$ to agent i. Repeat this procedure for all bundles. Finally, distribute any remaining intervals to any of these bundles arbitrarily. Let the resulting partition be $\mathcal{T}\simeq$.

By the end of these procedures, each bundle in $\mathcal{T}\simeq$ is worth at least $(1 - \frac{\epsilon}{2}) \cdot \mathrm{MMS}_i(n, M \cup C)$. Then by the definition of MMS, the second inequality holds. We remark that these steps are not actually implemented in our algorithm. They are only used to demonstrate the difference of MMS values for the two instances.

Now, because I' is a problem instance with only indivisible goods, we can compute a partition $(P_1, P_2, \ldots, P_n)$ such that $\min_{j \in N} u_i(P_j) \geq (1 - \frac{\epsilon}{2}) \cdot \mathrm{MMS}_i(n, M \cup \tilde{C})$ via the PTAS from [58] with $\delta = \epsilon/2$. It then holds that

$$\begin{aligned} \min_{j \in N} u_i(P_j) &\geq \left(1 - \frac{\epsilon}{2}\right)\left(1 - \frac{\epsilon}{2}\right) \cdot \mathrm{MMS}_i(n, M \cup C) \\ &\geq (1 - \epsilon)\mathrm{MMS}_i(n, M \cup C). \end{aligned}$$

The proof of Lemma 4 is complete. □

Lemma 4 also implies that in the mixed goods setting, we can compute in polynomial time a value MMS_i' such that $\mathrm{MMS}_i \geq \mathrm{MMS}_i' \geq (1 - \epsilon)\mathrm{MMS}_i$.

4.3.2 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 3 shows that the new algorithm with these approximate values will still terminate.

According to Lemma 4, 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.

4.3.3 Time complexity analysis

We start with the analyses of Robertson-Webb queries. In Algorithm 2, we need $O(n)$ value queries to obtain agents' values for the (heterogeneous) cake. In order to compute the weights, we also need $O(n)$ value queries to get agents' values for their pieces after running Mixed-MMS-Homogeneous (Algorithm 1). Then, in Algorithm 1, in each `while`-loop, we need $O(n)$ cut queries to find the leftmost cut point (line 10); overall, it takes $O(n^2)$ RW queries.

In light of Lemma 4, computing approximate MMS values takes polynomial time. Then the only step that needs 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 [30] 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 [30], WPRAlloc at line 4 of Algorithm 2 can be implemented in polynomial time. Summarize everything together, we obtain a polynomial-time algorithm.

Lemma 5 *Algorithm 2 runs in polynomial time with $O(n^2)$ Robertson–Webb queries and one call to the* WPRAlloc *oracle.*

Sections 4.1, 4.2 and 4.3 together complete the proof of Theorem 3.

4.4 Boosting the approximation ratio

In Theorem 3, 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 [38]. 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 3 with Corollary 1 ($\gamma_I = \gamma_M$), we can improve the ratio directly to $\max\{\alpha, \gamma_I\}$ in Theorem 3. 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 3 and compare it with β: if $\alpha' \geq \beta$, we directly apply Theorem 3; 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 *A* $\max\{\alpha, \gamma_I\}$*-MMS allocation with mixed goods always 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 utilizes the proof of Lemma 4 and is straightforward to prove. The currently best lower bound of γ_I is $\frac{3}{4} + \frac{1}{12n}$ and the currently best-known value of β is $\frac{3}{4}$, both are due to Garg and Taki [38]. 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, envy-freeness is often relaxed to *envy-freeness up to one item (EF1)* and MMS is often considered as a relaxation of proportionality. It is known from [28] that neither EF1 nor MMS implies the other. In a recent work, Bei et al. [16] 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 EF, EF1, and EFM here for the sake of being self-contained.

Definition 6 (*EF*) An allocation $\mathcal{A}$ is said to satisfy *envy-freeness (EF)* if for any pair of agents $i, j \in N$, $u_i(A_i) \geq u_i(A_j)$.

Definition 7 (*EF1*) With indivisible goods, an allocation $\mathcal{A}$ is said to satisfy *envy-freeness up to one good (EF1)* if $\forall i, j \in N$, $\exists g \in A_j$, such that $u_i(A_i) \geq u_i(A_j \setminus \{g\})$.

Definition 8 (*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 et al. [5] 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 6 *Given any mixed goods instance* $\langle N, M \cup C \rangle$, *for any EFM allocation* $(A_1, A_2, \ldots, A_n)$ *and any agent* $i \in N$, *we have*

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

The proof is a direct generalization of the proof of Proposition 3.6 in [5].

From Lemma 6, 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 Sect. 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 study the extent to which we can find approximate MMS allocations when the resources contain both divisible and indivisible goods. We analyze the relation of the worst-case MMS approximation guarantees between mixed goods instances and indivisible goods instances. We also present 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. It would also be interesting to investigate other fairness notions in the mixed goods setting—notions we mentioned like pairwise MMS and groupwise MMS fit the setting well. Another direction is to study fair allocations in the mixed goods setting in conjunction with economic efficiency notions such as Pareto optimality. There have been some preliminary results, for instance, in the context of determining the compatibility of EFM and PO by Bei et al. [16]. One could also further generalize the mixed resources model to capture other practical scenarios. One natural consideration is to let agents have either positive or negative utility for each item. This has been done, for example, when items are all divisible or indivisible [10, 19, 29].

References

1. Abdulkadiroğlu, A., Sönmez, T., & Ünver, M. U. (2004). Room assignment-rent division: A market approach. *Social Choice and Welfare, 22*(3), 515–538.
2. Alkan, A., Demange, G., & Gale, D. (1991). Fair allocation of indivisible goods and criteria of justice. *Econometrica, 59*(4), 1023–1039.
3. Alon, N. (1987). Splitting necklaces. *Advances in Mathematics, 63*(3), 247–253.
4. Amanatidis, G., Markakis, E., Nikzad, A., & Saberi, A. (2017). Approximation algorithms for computing maximin share allocations. *ACM Transactions on Algorithms, 13*(4), 52:1-52:28.
5. Amanatidis, G., Birmpas, G., & Markakis, E. (2018). Comparing approximate relaxations of envy-freeness. In *Proceedings of the 27th international joint conference on artificial intelligence (IJCAI)* (pp. 42–48).
6. Arunachaleswaran, E. R., Barman, S., & 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)* (pp. 1994–2013).
7. Aziz, H. (2021). Achieving envy-freeness and equitability with monetary transfers. In *Proceedings of the 35th AAAI conference on artificial intelligence (AAAI)* (pp. 5102–5109).

8. Aziz, H., & Mackenzie, S. (2016). A discrete and bounded envy-free cake cutting protocol for any number of agents. In *Proceedings of the 57th IEEE annual symposium on foundations of computer science (FOCS)* (pp. 416–427).
9. Aziz, H., Rauchecker, G., Schryen, G., & Walsh, T. (2017). Algorithms for max-min share fair allocation of indivisible chores. In *Proceedings of the 31st AAAI conference on artificial intelligence (AAAI)* (pp. 335–341).
10. Aziz, H., Caragiannis, I., Igarashi, A., & Walsh, T. (2019a). Fair allocation of indivisible goods and chores. In *Proceedings of the 28th international joint conference on artificial intelligence (IJCAI)* (pp. 53–59).
11. Aziz, H., Chan, H., & Li, B. (2019b). Weighted maxmin fair share allocation of indivisible chores. In *Proceedings of the 28th international joint conference on artificial intelligence (IJCAI)* (pp. 46–52).
12. Aziz, H., Li, B., & Wu, X. (2019c). Strategyproof and approximately maxmin fair share allocation of chores. In *Proceedings of the 28th international joint conference on artificial intelligence (IJCAI)* (pp. 60–66).
13. Barbanel, J. B. (2005). *The geometry of efficient fair division*. Cambridge: Cambridge University Press.
14. Barman, S., & Krishnamurthy, S. K. (2020). Approximation algorithms for maximin fair division. *ACM Transactions on Economics and Computation (TEAC), 8*(1), 5:1-5:28.
15. Barman, S., Biswas, A., Krishnamurthy, S. K., & Narahari, Y. (2018). Groupwise maximin fair allocation of indivisible goods. In *Proceedings of the 32nd AAAI conference on artificial intelligence (AAAI)* (pp. 917–924).
16. Bei, X., Li, Z., Liu, J., Liu, S., & Lu, X. (2021a). Fair division of mixed divisible and indivisible goods. *Artificial Intelligence, 293* 103436.
17. Bei, X., Liu, S., Lu, X., & Wang, H. (2021b). Maximin fairness with mixed divisible and indivisible goods. In *Proceedings of the 35th AAAI conference on artificial intelligence (AAAI)* (pp. 5167–5175).
18. Bhaskar, U., Sricharan, A., & Vaish, R. (2020). On approximate envy-freeness for indivisible chores and mixed resources. CoRR abs/2012.06788, arXiv:abs/2012.06788.
19. Bogomolnaia, A., Moulin, H., Sandomirskiy, F., & Yanovskaya, E. (2017). Competitive division of a mixed manna. *Econometrica, 85*(6), 1847–1871.
20. Bouveret, S., & Lemaître, M. (2016). Characterizing conflicts in fair division of indivisible goods using a scale of criteria. *Autonomous Agents and Multi-Agent Systems, 30*(2), 259–290.
21. Bouveret, S., Cechlárová, K., Elkind, E., Igarashi, A., & Peters, D. (2017). Fair division of a graph. In *Proceedings of the 26th international joint conference on artificial intelligence (IJCAI)* (pp. 135–141).
22. Brams, S. J. (2008). *Mathematics and democracy: Designing better voting and fair-division procedures*. Princeton University Press.
23. Brams, S. J., & Taylor, A. D. (1996). *Fair division: From cake-cutting to dispute resolution*. Cambridge University Press.
24. Brandt, F., Conitzer, V., Endriss, U., Lang, J., & Procaccia, A. D. (eds). (2016). *Handbook of computational social choice*. Cambridge University Press.
25. Brustle, J., Dippel, J., Narayan, V., Suzuki, M., & Vetta, A. (2020). One dollar each eliminates envy. In *Proceedings of the 21st ACM conference on economics and computation (EC)* (pp. 23–39).
26. Budish, E. (2011). The combinatorial assignment problem: Approximate competitive equilibrium from equal incomes. *Journal of Political Economy, 119*(6), 1061–1103.
27. Caragiannis, I., & Ioannidis, S. (2020). Computing envy-freeable allocations with limited subsidies. CoRR abs/2002.02789, arXiv:abs/2002.02789.
28. Caragiannis, I., Kurokawa, D., Moulin, H., Procaccia, A. D., Shah, N., & Wang, J. (2019). The unreasonable fairness of maximum Nash welfare. *ACM Transactions on Economics and Computation, 7*(3), 12:1-12:32.
29. Chaudhury, B. R., Garg, J., McGlaughlin, P., & Mehta, R. (2021). Competitive allocation of a mixed manna. In *Proceedings of the 32nd ACM-SIAM symposium on discrete algorithms (SODA)* (pp. 1405–1424).
30. Cseh, A., & Fleiner, T. (2020). The complexity of cake cutting with unequal shares. *ACM Transactions on Algorithms (TALG), 16*(3), 29:1-29:21.
31. Dubins, L. E., & Spanier, E. H. (1961). How to cut a cake fairly. *The American Mathematical Monthly, 68*(1), 1–17.
32. Edmonds, J., & Pruhs, K. (2011). Cake cutting really is not a piece of cake. *ACM Transactions on Algorithms (TALG), 7*(4), 51:1-51:12.

33. Endriss, U. (ed). (2017). *Trends in computational social choice*. AI Access.
34. Even, S., & Paz, A. (1984). A note on cake cutting. *Discrete Applied Mathematics, 7*(3), 285–296.
35. Farhadi, A., Ghodsi, M., Hajiaghayi, M., Lahaie, S., Pennock, D., Seddighin, M., et al. (2019). Fair allocation of indivisible goods to asymmetric agents. *Journal of Artificial Intelligence Research (JAIR), 64*(1), 1–20.
36. Foley, D. K. (1967). Resource allocation and the public sector. *Yale Economics Essays, 7*(1), 45–98.
37. Gal, Y. K., Mash, M., Procaccia, A. D., & Zick, Y. (2017). Which is the fairest (rent division) of them all? *Journal of the ACM (JACM), 64*(6), 39:1-39:22.
38. Garg, J., & Taki, S. (2020). An improved approximation algorithm for maximin shares. In *Proceedings of the 21st ACM conference on economics and computation (EC)* (pp. 379–380).
39. Garg, J., McGlaughlin, P., & Taki, S. (2019). Approximating maximin share allocations. In *Proceedings of the 2nd symposium on simplicity in algorithms (SOSA)* (pp. 20:1–20:11).
40. Ghodsi, M., Hajiaghayi, M., Seddighin, M., Seddighin, S., & Yami, H. (2020). Fair allocation of indivisible goods: Improvement. *Mathematics of Operations Research*. https://doi.org/10.1287/moor.2020.1096.
41. Goldman, J., & Procaccia, A. D. (2015). Spliddit: Unleashing fair division algorithms. *SIGecom Exchanges, 13*(2), 41–46.
42. Gourvès, L., & Monnot, J. (2019). On maximin share allocations in matroids. *Theoretical Computer Science, 754,* 50–64.
43. Haake, C. J., Raith, M. G., & 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.
44. Halpern, D., & Shah, N. (2019). Fair division with subsidy. In *Proceedings of the 12th international symposium on algorithmic game theory (SAGT)* (pp. 374–389).
45. Huang, X., & Lu, P. (2019). An algorithmic framework for approximating maximin share allocation of chores. In *Proceedings of the 22nd ACM conference on economics and computation (EC)* (forthcoming).
46. Igarashi, A., & Peters, D. (2019). Pareto-optimal allocation of indivisible goods with connectivity constraints. In *Proceedings of the 33rd AAAI conference on artificial intelligence (AAAI)* (pp. 2045–2052).
47. 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.
48. Kurokawa, D., Procaccia, A. D., & Wang, J. (2018). Fair enough: Guaranteeing approximate maximin shares. *Journal of the ACM (JACM), 65*(2), 8:1-8:27.
49. Lipton, R. J., Markakis, E., Mossel, E., & Saberi, A. (2004). On approximately fair allocations of indivisible goods. In *Proceedings of the 5th ACM Conference on electronic commerce (EC)* (pp. 125–131).
50. Lonc, Z., & Truszczynski, M. (2020). Maximin share allocations on cycles. *Journal of Artificial Intelligence Research (JAIR), 69,* 613–655.
51. 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*. Palgrave Macmillan UK, chap 11 (pp. 341–349).
52. Meertens, M., Potters, J., & Reijnierse, H. (2002). Envy-free and Pareto efficient allocations in economies with indivisible goods and money. *Mathematical Social Sciences, 44*(3), 223–233.
53. Moulin, H. (2019). Fair division in the internet age. *Annual Review of Economics, 11*(1), 407–441.
54. Robertson, J., & Webb, W. (1998). *Cake-cutting algorithm: Be fair if you can*. A K Peters/CRC Press.
55. Steinhaus, H. (1949). Sur la division pragmatique. *Econometrica, 17*(Supplement), 315–319.
56. Su, F. E. (1999). Rental harmony: Sperner's lemma in fair division. *The American Mathematical Monthly, 106*(10), 930–942.
57. Suksompong, W. (2018). Approximate maximin shares for groups of agents. *Mathematical Social Sciences, 92,* 40–47.
58. Woeginger, G. J. (1997). A polynomial-time approximation scheme for maximizing the minimum machine completion time. *Operations Research Letters, 20*(4), 149–154.
59. Young, H. P. (1995). *Equity. In theory and practice*. Princeton: Princeton University Press.