Breaking the 3/4 Barrier for Approximate Maximin Share*

Hannaneh Akrami† Jugal Garg‡

Abstract

We study the fundamental problem of fairly allocating a set of indivisible goods among n agents with additive valuations using the desirable fairness notion of maximin share (MMS). MMS is the most popular share-based notion, in which an agent finds an allocation fair to her if she receives goods worth at least her MMS value. An allocation is called MMS if all agents receive at least their MMS value. However, since MMS allocations need not exist when $n > 2$, a series of works showed the existence of approximate MMS allocations with the current best factor of $\frac{3}{4} + O(\frac{1}{n})$. The recent work [3] showed the limitations of existing approaches and proved that they cannot improve this factor to $3/4 + \Omega(1)$. In this paper, we bypass these barriers to show the existence of $(\frac{3}{4} + \frac{3}{3836})$-MMS allocations by developing new reduction rules and analysis techniques.

1 Introduction

Fair allocation of resources (goods) is a fundamental problem in the intersection of computer science, economics, and social choice theory. This age-old problem arises naturally in a wide range of real-life settings, which was formally introduced in the seminal work of Steinhaus in the 1940s [44]. Depending on what properties the goods have and what notion of fairness is considered, one can address a wide range of problems. Extensive work has been done for the case of *divisible* goods, where goods can be fractionally allocated, e.g., [47, 30, 10, 11].

More recently, fair division of indivisible goods has received significant attention due to their applications in various multi-agent settings. Formally, an instance of fair division of indivisible goods consists of a set $N = \{1, 2, \ldots, n\}$ of agents, a set M of m indivisible goods, and valuation vector $\mathcal{V} = (v_1, \ldots, v_n)$ where $v_i : 2^M \to \mathbb{R}_{\geq 0}$ is the valuation function of agent i. The goal is to find an allocation $A = \langle A_1, A_2, \ldots, A_n \rangle$, in which agent i gets A_i, and A satisfies some fairness criteria.

Two main categories of fairness are envy-based notions and share-based notions. Roughly speaking, in envy-based notions, an agent finds an allocation fair by comparing her bundle with other agents' bundles. Under allocation A, if certain conditions are met for all agents (e.g., $v_i(A_i) \geq v_i(A_j)$ for all $i, j \in N$ in the case of envy-freeness), then A is fair. Popular examples of envy-based notions are envy-freeness (EF) and its relaxations envy-freeness up to any good (EFX) [24], and envy-freeness up to one good (EF1) [42].

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

In this paper, we consider a natural relaxation of proportionality called *maximin share (MMS)*, introduced by Budish [23]. It is also preferred by participating agents over other notions, as shown in real-life experiments by [33]. Maximin share of an agent is the maximum value she can guarantee to obtain if she divides the goods into n bundles (one for each agent) and receives a bundle with the minimum value. Basically, for an agent i, assuming that all agents have i's valuation function, the maximum value one can guarantee for all the agents is

*Jugal Garg was supported by NSF Grant CCF-1942321.

†Max Planck Institute for Informatics and Graduiertenschule Informatik, Universität des Saarlandes, Germany.
‡University of Illinois at Urbana-Champaign, USA.

Copyright © 2024
Copyright for this paper is retained by authors

	Existence	**Non-existence**
$n = 3$	$11/12$ [28]	$> 39/40$ [29]
$n = 4$	$4/5$ [34]	$> 1 - 4^{-4}$ [29]
$n > 4$	$2/3$ [43, 8, 40, 31] $2/3(1 + 1/(3n - 1))$ [19] $3/4$ [34] $3/4 + 1/(12n)$ [32] $3/4 + \min(1/36, 3/(16n - 4))$ [3] $3/4 + 3/3836$ **(Theorem 6.1)**	$> 1 - \mathcal{O}(\frac{1}{2^n})$ [43] $> 1 - \frac{1}{n^4}$ [29]

Table 1: Summary of the approximate MMS results when agents have additive valuations

the i's maximin share, denoted by MMS_i. Formally, for a set S of goods and any positive integer d, let $\Pi_d(S)$ denote the set of all partitions of S into d bundles. Then,

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

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

Since MMS allocations do not always exist when there are three or more agents with additive valuations [43, 29], the focus shifted to study approximations of MMS. An allocation A is α-MMS if $v_i(A_i) \geq \alpha \cdot \text{MMS}_i$ for all agents $i \in N$. We note that the MMS notion is closely related to the popular max-min objective or the classic Santa Claus problem ($\max_A \min_i v_i(A_i)$) [16]. Unlike the max-min objective, the (α-)MMS objective satisfies the desirable scale-invariance property. In the case of agents with identical valuations, an exact MMS allocation exists, and in this case, finding α-MMS allocation is equivalent to α-approximation of the Santa Claus problem. The best approximation factor known for the max-min objective under additive valuations is $\tilde{O}(m^{\varepsilon})$ for any $\varepsilon > 0$ [25].

For the MMS problem, Procaccia and Wang [43] showed the existence of 2/3-MMS allocations. Many follow-up works have improved the approximation factor [19, 34, 31, 8, 40, 32] with the current best result of $\alpha = \frac{3}{4} + \min(\frac{1}{36}, \frac{3}{16n-4})$ [3]. However, since the work of Ghodsi et al. [34], the best known constant approximation factor for MMS has remained 3/4 for large n. In this work, we break this 3/4 wall by proving the existence of $(\frac{3}{4} + \frac{3}{3836})$-MMS allocations.

After Ghodsi et al. [34] proved the existence of 3/4-MMS allocations and gave a PTAS to compute one, Garg and Taki [32] gave a simple algorithm with complicated analysis proving the existence of $(\frac{3}{4} + \frac{1}{12n})$-MMS allocations and also computing a 3/4-MMS allocation in polynomial time. Very recently, Akrami et al. [3] simplified the analysis of (a slight modification of) the Garg-Taki algorithm significantly and proved the existence of $(\frac{3}{4} + \min(\frac{1}{36}, \frac{3}{16n-4}))$-MMS allocations. Moreover, they gave a tight example for this algorithm showing that no constant factor better than 3/4 can be obtained for approximate MMS using this approach. In Section 3, we discuss the known techniques' barriers in more detail and how our algorithm overcomes these barriers.

The complementary problem is to find upper bounds on the largest α for which α-MMS allocations exist. Feige et al. [29] constructed an example with three agents and nine goods for which no allocation is better than 39/40-MMS. For $n \geq 4$, their construction gives an example for which no allocation is better than $(1 - n^{-4})$-MMS. Table 1 summarizes all these results. We note that most of these existence results can be easily converted into PTAS for finding such an allocation using the PTAS for finding the MMS values [48].

1.1 Further related work Special cases. There has been a line of work on the instances with a limited number of agents or goods. When $m \leq n + 3$, an MMS allocation always exists [8]. Feige et al. [29] improved this bound to $m \leq n + 5$. For $n = 2$, MMS allocations always exist [22]. For $n = 3$, the MMS approximation was improved from 3/4 [43] to 7/8 [8] to 8/9 [35], and then to 11/12 [28]. For $n = 4$, Ghodsi et al. [34] showed the existence of 4/5-MMS. For $n \geq 5$, the best known factor is the general $(\frac{3}{4} + \min(\frac{1}{36}, \frac{3}{16n-4}))$ bound given by Akrami et al. [3].

Ordinal approximation. An alternative way of relaxing MMS is guaranteeing 1-out-of-d maximin share (MMS)

Copyright © 2024
Copyright for this paper is retained by authors

for $d > n$, which is the maximum value that an agent can ensure by partitioning the goods into d bundles and choosing the least preferred bundle. This notion only depends on the bundles' ordinal ranking and is not affected by a small perturbation in the value of every single good (as long as the ordinal ranking of the bundles does not change). A series of works studied this notion [1, 36, 37] with the state-of-the-art being the existence of 1-out-of-$\lceil \frac{4n}{3} \rceil$ MMS allocations for goods [4].

Chores. MMS can be analogously defined for fair division of chores. MMS allocations do not always exist for chores [12], which motivated the study of approximate MMS [12, 19], with the current best approximation ratio being very recently improved from 11/9 [38] to 13/11 [39]. In the case of $n = 3$, 19/18-MMS allocations exist [28].

MMS in the chores setting is closely related to the well-studied variants of bin-packing and job scheduling problems. In particular, the recent paper [39] utilizes the Multifit algorithm for makespan minimization to obtain the best approximation factor. Therefore, many ideas which are already developed are proven to be useful when dealing with chores. On the other hand, when dealing with goods, the related variants of bin packing and scheduling problems do not make much sense where the objective becomes to maximize the number/capacity of bins or maximize the minimum processing time of a machine while allocating all the items. Therefore, new ideas specific to this problem are required. Furthermore, although the explicit study of MMS for goods started much before chores, the advancement in approximate MMS for chores has been faster. Also, the current best factor (13/11) is much better than the analogous factor for goods $(3/4 + 3/3836)$, despite the extensive work by many researchers on the goods problem. For ordinal approximation, the best-known factor for existence is 1-out-of-$\lfloor \frac{3n}{4} \rfloor$ MMS allocations for chores.

Other settings. The MMS notion has also been studied when agents have more general valuations than additive, e.g., [19, 34, 41, 46, 5]. Generalizations have also been studied where restrictions are imposed on the set of feasible allocations, such as matroid constraints [35], cardinality constraints [21], and graph connectivity constraints [20, 45, 49]. Strategyproof versions of fair division have also been studied [18, 7, 6, 9]. MMS has also inspired other notions of fairness, like weighted MMS [27], AnyPrice Share (APS) [13], pairwise MMS [24], groupwise MMS [17, 26], 1-out-of-d share [36], and self-maximizing shares [15]. MMS has also been studied in best-of-both-worlds settings, where both ex-ante and ex-post guarantees are sought [14, 5].

2 Preliminaries

For all $n \in \mathbb{N}$, let $[n] = \{1, 2, \ldots, n\}$. A fair division instance $\mathcal{I} = (N, M, \mathcal{V})$ consists of a set of agents $N = [n]$, a set of goods $M = [m]$ and a vector of valuation functions $\mathcal{V} = (v_1, v_2, \ldots, v_n)$ such that for all $i \in [n]$, $v_i : 2^M \to \mathbb{R}_{\geq 0}$ indicates how much agent i likes each subset of the goods. In this paper, we assume the valuation functions are additive, i.e., for all $i \in [n]$ and $S \subseteq M$, $v_i(S) = \sum_{g \in S} v_i(\{g\})$. For ease of notation, for all $g \in M$, we use $v_i(g)$ or $v_{i,g}$ instead of $v_i(\{g\})$.

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

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

When the instance $\mathcal{I} = (N, M, \mathcal{V})$ is clear from the context, we denote $\text{MMS}^n_{v_i}$ by $\text{MMS}_i(\mathcal{I})$ or MMS_i for all $i \in [n]$. For each agent i, let $P^i = (P^i_1, P^i_2, \ldots, P^i_n)$ be a partition of M into n bundles admitting the MMS value of agent i. Formally, $\text{MMS}_i = \min_{j \in [n]} v_i(P^i_j)$. We call such a partition, an MMS partition of agent i. An allocation X is MMS if for all agents $i \in N$, $v_i(X_i) \geq \text{MMS}_i$. Similarly, for any $0 < \alpha \leq 1$, an allocation X is α-MMS if $v_i(X_i) \geq \alpha \cdot \text{MMS}_i$ for all agents $i \in N$.

Definition 2.1. (Ordered instance) *An instance $\mathcal{I} = (N, M, \mathcal{V})$ is ordered if there exists an ordering of the goods $(g_1, g_2, \ldots, g_m)$ such that for all agents $i \in N$, $v_i(g_1) \geq v_i(g_2) \geq \ldots \geq v_i(g_m)$.*

It is known that the hardest instances of approximating MMS are the ordered instances [19]. We use the notations used in [3].

Definition 2.2. ([3]) *For the fair division instance $\mathcal{I} = ([n], [m], \mathcal{V})$, $\texttt{order}(\mathcal{I})$ is defined as the instance $([n], [m], \mathcal{V}')$, where for each $i \in [n]$ and $j \in [m]$, $v'_i(j)$ is the j^{th} largest number in the multiset $\{v_i(g) \mid g \in [m]\}$.*

Copyright © 2024
Copyright for this paper is retained by authors

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

for $i \in N$ **do**
  Compute agent i's MMS partition P^i.
  $\forall j \in N$, $\forall g \in P^i_j$, let $v'_{i,g} \leftarrow v_{i,g}/v_i(P^i_j)$.
end for
return $(N, M, \mathcal{V}')$.

The transformation `order` is α-*MMS-preserving*, i.e., for a fair division instance $\mathcal{I}$, given an α-MMS allocation of `order`$(\mathcal{I})$, one can compute an α-MMS allocation of $\mathcal{I}$ in polynomial time [19]. Given any ordered instance $\mathcal{I} = ([n], [m], \mathcal{V})$, without loss of generality, we assume $v_i(1) \geq v_i(2) \geq \ldots \geq v_i(m)$ for all $i \in [n]$.

Lemma 2.1. ([19]) *Given an instance $\mathcal{I}$ and an α-MMS allocation of* `order`$(\mathcal{I})$*, one can compute an α-MMS allocation of $\mathcal{I}$ in polynomial time.*

Definition 2.3. (Normalized instance) *An instance $\mathcal{I} = (N, M, \mathcal{V})$ is* normalized*, if for all $i, j \in [n]$, $v_i(P^i_j) = 1$.*

Note that since v_i is additive, if $\mathcal{I}$ is normalized, then for all MMS partitions of i like $Q = (Q_1, \ldots, Q_n)$ and for all $j \in [n]$ we have $v_i(Q_j) = 1$. [3] shows that given any instance $\mathcal{I} = (N, M, \mathcal{V})$, one can compute a normalized instance $\mathcal{I}' = (N, M, \mathcal{V}')$ such that any α-MMS allocation for $\mathcal{I}'$ is an α-MMS allocation for $\mathcal{I}$. Their algorithm converting an instance to a normalized instance is shown in Algorithm 1. We note that since finding an agent's MMS value is NP-hard, this is not a polynomial-time algorithm, but a PTAS exists.

Lemma 2.2. ([3]) *Let $\mathcal{I}' = (N, M, \mathcal{V}') =$* `normalize`$(\mathcal{I} = (N, M, \mathcal{V}))$*. Then for any allocation A, $v_i(A_i) \geq v'_i(A_i)\mathrm{MMS}_i(\mathcal{I})$ for all $i \in N$.*

Lemma 2.2 implies that `normalize` is α-MMS-preserving, since if A is an α-MMS allocation for the normalized instance $(N, M, \mathcal{V}')$, then A is also an α-MMS allocation for the original instance $(N, M, \mathcal{V})$. [3] give some structural property of ordered normalized instances which we repeat here in Lemma 2.3.

Lemma 2.3. *[3] Let $([n], [m], \mathcal{V})$ be an ordered and normalized fair division instance. For all $k \in [n]$ and agent $i \in [n]$, if $v_i(k) + v_i(2n - k + 1) > 1$, then $v_i(2n - k + 1) \leq 1/3$ and $v_i(k) > 2/3$.*

2.1 Reduction rules Given any instance $\mathcal{I}$, a reduction rule $R(\mathcal{I})$ is a procedure that allocates a subset $S \subseteq M$ of goods to an agent i and outputs the instance $\mathcal{I}' = (N \setminus \{i\}, M \setminus S, \mathcal{V})$.

Definition 2.4. (Valid reductions) *Let R be a reduction rule and $R(\mathcal{I}) = (N', M', \mathcal{V})$ such that $\{i\} = N \setminus N'$ and $S = M \setminus M'$. Then R is a "valid α-reduction" if*

1. *$v_i(S) \geq \alpha \cdot MMS^{|N|}_{v_i}(M)$, and*
2. *for all $j \in N'$, $MMS^{|N|-1}_{v_j}(M') \geq MMS^{|N|}_{v_j}(M)$.*

Furthermore, a reduction rule R is a "valid reduction for agent $j \in N'$", if $MMS^{|N|-1}_{v_j}(M') \geq MMS^{|N|}_{v_j}(M)$ where N' and M' are the set of remaining agents and remaining goods respectively after the reduction.

Note that if R is a valid α-reduction and an α-MMS allocation A exists for $R(\mathcal{I})$, then an α-MMS allocation exists for $\mathcal{I}$. Such an allocation can be obtained by allocating S to i and allocating the rest of the goods as they are allocated under A.

Lemma 2.4. *Given an instance $\mathcal{I} = (N, M, \mathcal{V})$, let $S \subseteq M$ be such that $v_i(S) \leq MMS_i$ and $|S| \leq 2$. Then allocating S to an arbitrary agent $j \neq i$, is a valid reduction for agent i.*

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

Definition 2.5. *For an ordered instance $\mathcal{I} = (N, M, \mathcal{V})$ and $\alpha > 0$, reduction rules R^α_1, R^α_2, R^α_3 and R^α_4 are defined as follows.*

Copyright © 2024
Copyright for this paper is retained by authors

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

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

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

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

We note that R_1^α, R_2^α, R_4^α in addition to one more rule of allocating $\{n, n+1\}$ to an agent is used in [32, 3]. Our algorithm does not use the rule of allocating $\{n, n+1\}$. Moreover, R_3^α (allocating $\{3n-2, 3n-1, 3n, 3n+1\}$) is used in our work and not elsewhere.

Lemma 2.5. *Given any* $\alpha > 0$ *and an ordered instance* $\mathcal{I}$*,* R_1^α*,* R_2^α*, and* R_3^α *are valid reductions for all the remaining agents.*

Proposition 2.1. *If* $\mathcal{I}$ *is ordered and for a given* $\alpha \geq 0$*, none of the rules* R_1^α*,* R_2^α *or* R_3^α *is applicable, then*

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

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

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

Proof. We prove each case separately.

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

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

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

□

Definition 2.6. (α-irreducible and δ-ONI) *We call an instance* $\mathcal{I}$ α*-irreducible if none of the rules* R_1^α*,* R_2^α*,* R_3^α *or* R_4^α *is applicable. Moreover, we call an instance* δ*-ONI if it is ordered, normalized, and* $(3/4+\delta)$*-irreducible.*

3 Technical overview

Most algorithms for approximating MMS, especially those with a factor of at least 3/4 [34, 32, 3], utilize two simple tools: valid reductions and bag filling. Although these tools are easy to use in a candidate algorithm, the novelty of these works is in the analysis, which is challenging. Like previous works, the analysis is the most difficult part of our algorithm based on these tools. Unlike previous works, we also need to use a new reduction rule and initialize bags differently, which are counterintuitive.

First, we discuss the algorithm given by [3], which is a slight modification of the algorithm in [32]. For $\alpha \leq 3/4$, [3] showed how to obtain an ordered normalized α-irreducible instance from any arbitrary instance such that the transformation is α-MMS preserving.* That is, given an α-MMS allocation for the resulting ordered normalized irreducible instance, one can obtain an α-MMS allocation for the original instance. In the first phase of their algorithm, they obtain an ordered normalized α-irreducible instance $\hat{\mathcal{I}}$ and in the second phase, they compute an α-MMS allocation for $\hat{\mathcal{I}}$. Let $\hat{\mathcal{I}} = ([n], [m], \mathcal{V})$. Without loss of generality, we can assume that $m \geq 2n$ (Observation 5.1).

In the second phase, they initialize n bags with the first $2n$ goods as follows.

$$B_k := \{k, 2n-k+1\} \text{ for } k \in [n] \tag{3.2}$$

Copyright © 2024
Copyright for this paper is retained by authors

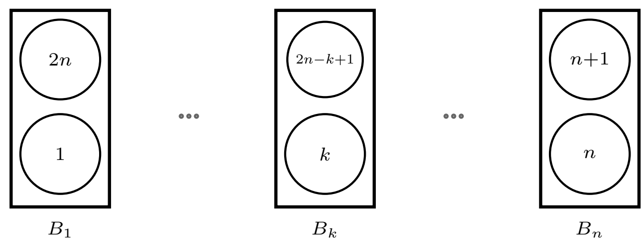

Figure 1: Configuration of Bags $B_1, B_2, \ldots, B_n$

Algorithm 2 `bagFill`$(\mathcal{I}, \alpha)$

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

```
for k ∈ [n] do
    B_k = {k, 2n + 1 − k}.
end for
U_G = [m] \ [2n]                                   ▷ unassigned goods
U_A = [n]                                          ▷ unsatisfied agents
U_B = [n]                                          ▷ unassigned bags
while U_A ≠ ∅ do
    if ∃i ∈ U_A, ∃k ∈ U_B, such that v_i(B_k) ≥ α then
        A_i = B_k
        U_A = U_A \ {i}
        U_B = U_B \ {k}
    else
        g = arbitrary good in U_G
        k = arbitrary bag in U_B
        B_k = B_k ∪ {g}.
        U_G = U_G \ {g}
    end if
end while
return ⟨A_1, ..., A_n⟩
```

See Figure 1 for a better intuition. As long as an agent i values a bag B_k at least α, allocate B_k to i and remove B_k and i. Then, as long as an unallocated bag exists (and thus a remaining agent), pick an arbitrary remaining bag B_k and add unassigned goods $g > 2n$ until some remaining agent i values it at least α. Then, allocate B_k to i and continue. The second phase is called the bag-filling phase. Algorithm 2 shows the pseudocode of the bag-filling phase of [3].

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

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

*[3] uses R_1^α, R_2^α, R_4^α and one more rule as reduction rules. However, all that matters in their proof is that the applied reduction rules are valid α-reduction rules.

Copyright © 2024
Copyright for this paper is retained by authors

Agents in N^2 The main bulk and difficulty of the analysis of [32] is to prove that all agents in N^2 receive a bag. By normalizing the instance, [3] managed to simplify this argument significantly. [3] prove $v_i(g) < 1/12$ for all $i \in N^2$ and all $g \in M \setminus [2n]$. This helps to bound the value of the bags that receive some goods in the bag-filling phase by $5/6$ for all available $i \in N^2$. Again, if the number of such bags is high enough, it is easy to prove that the algorithm does not run out of goods in the bag-filling phase. The difficult case is when the total value of the bags which are of value more than 1 to some agent $i \in N^2$ is large. Roughly speaking, in this case, it seems that the bags which receive goods in the bag-filling phase and their values are bounded by $5/6$ cannot compensate for the large value of the bags that do not require any goods in the bag-filling phase. This is where the normalized property of $\hat{\mathcal{I}}$ simplifies the matter significantly. Intuitively, there are many goods with a high value that happened to be paired in the same bag in the bag initialization phase. Since the instance is normalized, we know that in the MMS partition of i, these goods cannot be in the same bag. This implies that many bags in the MMS partition of i have at most 1 good in common with the goods in $[2n]$. This means that the value of the remaining goods (the goods in $M \setminus [2n]$) must be large since they fill the bags in the MMS partition such that the value of each bag equals 1. Hence, enough goods remain in $M \setminus [2n]$ to fill the bags.

There are two main obstacles to generalizing this algorithm to obtain α-MMS allocations when $\alpha > 3/4$. The first obstacle lies in the first phase of the algorithm. R_4^α is a valid α-reduction when $\alpha \leq 3/4$ and R_1^α and R_2^α are not applicable. This no longer holds when $\alpha > 3/4$. In this case, the MMS value of the agents can indeed decrease after applying R_4^α. When $\alpha = 3/4 + \mathcal{O}(1/n)$, [32] and [3] managed to resolve this issue by adding some dummy goods after each iteration of R_4^α and proving that the total value of these dummy goods is negligible. Essentially, since we only need to guarantee the last agent a value of α, the idea is to divide the excess $1 - \alpha$ among all agents and improve the factor. However, this can only improve the factor by at most $\mathcal{O}(1/n)$. If $\alpha > 3/4 + \epsilon$ for a constant $\epsilon > 0$, the same technique does not work since the value of dummy goods cannot be reasonably bounded. We resolve this issue in Section 4. Unlike the previous works, we allow the MMS values of the remaining agents to drop. Although the MMS values of the agents can drop, we show that they do not drop by more than a multiplicative factor of $(1 - 4\epsilon)$ after an arbitrary number of applications of $R_k^{3/4+\epsilon}$ for $k \in [4]$. Basically, while for $\alpha \leq 3/4$, one can get α-irreducibility for free (i.e., without losing any approximation factor on MMS), for $\alpha = 3/4 + \epsilon$ and $\epsilon > 0$, we lose an approximation factor of $(1 - 4\epsilon)$.

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

We break the problem into two cases depending on the number of these problematic agents. In Section 5.1, we consider the case when the number of problematic agents is not too large. In this case, we work with a slight modification of the algorithm in [3], and using an involved analysis, we show that it gives a $(3/4+\epsilon)$-MMS allocation. Otherwise, we introduce a new reduction rule for the first time that allocates the two most valuable goods to an agent. Although allocating these goods seem counterintuitive, surprisingly, that seems to be the only way to obtain a $(3/4 + \epsilon)$-MMS allocation for the tight example in [3]. In Section 5.2, we give another algorithm to handle the case where the number of problematic agents is too large. In this case, we first apply the reduction rules (including the new one), and then initialize the bags with three goods, unlike the previous works. Precisely, we set $C_k := \{k, 2n - k + 1, 2n + k\}$ and then do bag-filling.

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

In Section 5, we prove a $(3/4 + \delta)$-MMS allocation exists for all δ-ONI instances for any $\delta \leq 3/956$. The main results of Sections 4 and 5 imply that for $4\epsilon/(1 - 4\epsilon) \leq \delta \leq 3/956$, a $\min(3/4 + \epsilon, (3/4 + \delta)(1 - 4\epsilon))$-MMS exists for all instances. Setting $\delta = 3/956$ and $\epsilon = \delta/(4(\delta + 1)) = 3/3836$, there always exists a $(3/4 + 3/3836)$-MMS allocation. We give the formal proof in Section 6. All the missing proofs are available in the full version of the paper [2].

Copyright © 2024
Copyright for this paper is retained by authors

Algorithm 3 `reduce`$((N, M, \mathcal{V}), \epsilon)$

```
1: I ← order(N, M, V)
2: for i ∈ N do
3:     v_{i,g} ← v_{i,g}/MMS_i, ∀g ∈ [m]
4: end for
5: while R_1^{(3/4+ε)} or R_2^{(3/4+ε)} or R_3^{(3/4+ε)} or R_4^{(3/4+ε)} is applicable do
6:     I ← R_k^{(3/4+ε)}(I) for smallest possible k
7: end while
8: return I.
```

4 Reduction to δ-ONI instances

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

We start with transforming the instance into an ordered one using the `order` algorithm. Then we scale the valuations such that for all $i \in N$, $\text{MMS}_i = 1$. Then, as long as one of the reduction rules R_1, R_2, R_3, or R_4 is applicable, we apply R_k for the smallest possible k. Algorithm 3 shows the pseudocode of this procedure.

In this section, we prove the following two theorems.

THEOREM 4.1. *Given an instance $\mathcal{I} = (N, M, \mathcal{V})$ and $\epsilon \geq 0$, let $\mathcal{I}' = (N', M', \mathcal{V}') =$ `reduce`$(\mathcal{I}, \epsilon)$. For all agents $i \in N'$, $MMS_i(\mathcal{I}') \geq 1 - 4\epsilon$.*

THEOREM 4.2. *Given an instance $\mathcal{I}$ and $\epsilon \geq 0$, let $\hat{\mathcal{I}} =$ `order`(`normalize`(`reduce`$(\mathcal{I}, \epsilon)$)). Then $\hat{\mathcal{I}}$ is ordered, normalized and $(\frac{3}{4} + \frac{4\epsilon}{1-4\epsilon})$-irreducible ($\frac{4\epsilon}{1-4\epsilon}$-ONI). Furthermore, from any α-MMS allocation of $\hat{\mathcal{I}}$ one can obtain a $\min(3/4+\epsilon, (1-4\epsilon)\alpha)$-MMS allocation of $\mathcal{I}$.*

Note that once R_1 is not applicable, we have $v_i(1) < 3/4 + \epsilon$ for all remaining agents i. Since we never increase the values, R_1 can no longer apply. So `reduce`$(\mathcal{I}, \epsilon)$ first applies R_1 as long as it is applicable and then applies the rest of the reduction rules. Since R_1 is a valid reduction rule for all the remaining agents i by Lemma 2.5, $\text{MMS}_i \geq 1$ after applications of R_1. So to prove Theorem 4.1 without loss of generality, we assume R_1 is not applicable on $\mathcal{I} = ([n], M, \mathcal{V})$. Let $\mathcal{I}' = (N', M', \mathcal{V}) =$ `reduce`$(\mathcal{I}, \epsilon)$. For the rest of this section, we fix agent $i \in N'$. Let $P = (P_1, P_2, \ldots, P_n)$ be the initial MMS partition of i (in $\mathcal{I}$). We construct a partition $Q = (Q_1, Q_2, \ldots, Q_{|N'|})$ of M' such that $v_i(Q_j) \geq 1 - 4\epsilon$ for all $j \in [|N'|]$.

Let G_2, G_3, and G_4 be the set of goods removed by applications of R_2, R_3, and R_4, respectively. Also, let $r_2 = |G_2|/3$, $r_3 = |G_3|/4$, and $r_4 = |G_4|/2$ be the number of times each rule is applied, respectively. Note that in the end, all that matters is that we construct a partition Q of $M \setminus (G_2 \cup G_3 \cup G_4)$ into $n - (r_2 + r_3 + r_4)$ bundles of value at least $1 - 4\epsilon$ for i. For this sake, it does not matter in which order the goods are removed. Therefore, without loss of generality, we assume all the goods in G_4 are removed first, and then the goods in G_2 and G_3 are removed in their original order. Note that we are not applying the reduction rules in a different order. We are removing the same goods that would be removed by applying the reduction rules in their original order. Only for the sake of our analysis, we remove these goods in a different order. For better intuition, consider the following example. Assume `reduce`$(\mathcal{I}, \epsilon)$ first applies R_2 removing $\{a_1, a_2, a_3\}$, then R_4 removing $\{b_1, b_2\}$, then another R_2 removing $\{c_1, c_2, c_3\}$ and then R_3 removing $\{d_1, d_2, d_3, d_4\}$. Without loss of generality we can assume that first $\{b_1, b_2\}$ is removed, then $\{a_1, a_2, a_3\}$, then $\{c_1, c_2, c_3\}$ and then $\{d_1, d_2, d_3, d_4\}$.

We know that when there are n agents, removing $\{2n-1, 2n, 2n+1\}$ (or $\{3n-2, 3n-1, 3n, 3n+1\}$) and an agent is a valid reduction for i by Lemma 2.5. With the same argument, it is not difficult to see that removing $\{g_1, g_2, g_3\}$ where $g_1 \geq 2n-1$, $g_2 \geq 2n$ and $g_3 \geq 2n+1$ (or $\{g_1, g_2, g_3, g_4\}$ where $g_1 \geq 3n-2$, $g_2 \geq 3n-1$, $g_3 \geq 3n$ and $g_4 \geq 3n+1$) and an agent is also a valid reduction for i.

LEMMA 4.1. *Let $\mathcal{I} = (N, M, \mathcal{V})$ be an ordered instance and $i \in N$.*

Copyright © 2024
Copyright for this paper is retained by authors

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

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

Observation 4.1. *Given an ordered instance $\mathcal{I} = (N, M, \mathcal{V})$, let $v_i(g_1) \geq \ldots \geq v_i(g_m), \forall i \in N$. Let $\mathcal{I}' = (N', M', \mathcal{V})$ be the instance after removing an agent i and a set of goods $\{a, b\}$ from $\mathcal{I}$. Let $g \in M'$ be the j^{th} most valuable good in M and the ${j'}^{th}$ most valuable good in M'. Then $j' \geq j-2$.*

Corollary 4.1. (of Observation 4.1) *Given an ordered instance $\mathcal{I} = (N, M, \mathcal{V})$, let $\mathcal{I}' = (N', M', \mathcal{V})$ be the instance after removing an agent i and a set of goods $\{a, b\}$ from $\mathcal{I}$. Let $n = |N|$ and $n' = |N'| = n-1$. Let $g \in M'$ be the j^{th} most valuable good in M and the ${j'}^{th}$ most valuable good in M'. Then,*

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

Next, assume at a step where the number of agents is n, $\{g_{2n-1}, g_{2n}, g_{2n+1}\}$ (or $\{g_{3n-2}, g_{3n-1}, g_{3n}, g_{3n+1}\}$) is removed with an application of R_2 (or R_3). Corollary 4.1 together with Lemma 4.1 imply that removing $\{g_{2n-1}, g_{2n}, g_{2n+1}\}$ (or $\{g_{3n-2}, g_{3n-1}, g_{3n}, g_{3n+1}\}$) at a later step where the number of agents is $n' \leq n$ is also valid for agent i. Therefore, all that remains is to prove that after removing the goods in G_4 and r_4 agents, the MMS value of i remains at least $1-4\epsilon$. That is, $\text{MMS}^{n-r_4}_i(M \setminus G_4) \geq 1-4\epsilon$.

Lemma 4.2. *Let $(N', M', \mathcal{V}) = \texttt{reduce}(([n], M, \mathcal{V}), \epsilon)$. Let r_4 be the number of times R_4 is applied during $\texttt{reduce}(\mathcal{I}, \epsilon)$ and let G_4 be the set of removed goods by applications of R_4. Then for all agents $i \in N'$, $MMS^{n-r_4}_{v_i}(M \setminus G_4) \geq 1-4\epsilon$.*

We are ready to prove Theorem 4.1 and 4.2.

Theorem 4.1. *Given an instance $\mathcal{I} = (N, M, \mathcal{V})$ and $\epsilon \geq 0$, let $\mathcal{I}' = (N', M', \mathcal{V}') = \texttt{reduce}(\mathcal{I}, \epsilon)$. For all agents $i \in N'$, $MMS_i(\mathcal{I}') \geq 1-4\epsilon$.*

Proof. Fix an agent $i \in N'$. Let $\mathcal{I}^1$ be the instance after all applications of R_1 and before any further reduction. By Lemma 2.5, $\text{MMS}_i(\mathcal{I}^1) \geq 1$. So without loss of generality, let us assume $\mathcal{I} = \mathcal{I}^1$. Let G_2, G_3, and G_4 be the set of goods removed by applications of R_2, R_3, and R_4, respectively. Also, let $r_2 = |G_2|/3$, $r_3 = |G_3|/4$, and $r_4 = |G_4|/2$ be the number of times each rule is applied, respectively. By Lemma 4.2, $\text{MMS}^{n-r_4}_{v_i}(M \setminus G_4) \geq 1-4\epsilon$. For an application of R_2 (or R_3) at step t, let $\{a_1, a_2, a_3\}$ (or $\{b_1, b_2, b_3, b_4\}$) be the set of goods that are removed. By Lemma 4.1, removing this set at a step $t' \geq t$ is still a valid reduction for i. Therefore, removing G_2 and G_3 and $r_2 + r_3$ agents does not decrease the MMS value of i. Thus, $\text{MMS}_i(\mathcal{I}') \geq 1-4\epsilon$. □

Theorem 4.2. *Given an instance $\mathcal{I}$ and $\epsilon \geq 0$, let $\hat{\mathcal{I}} = \texttt{order}(\texttt{normalize}(\texttt{reduce}(\mathcal{I}, \epsilon)))$. Then $\hat{\mathcal{I}}$ is ordered, normalized and $(\frac{3}{4} + \frac{4\epsilon}{1-4\epsilon})$-irreducible ($\frac{4\epsilon}{1-4\epsilon}$-ONI). Furthermore, from any α-MMS allocation of $\hat{\mathcal{I}}$ one can obtain a $\min(3/4+\epsilon, (1-4\epsilon)\alpha)$-MMS allocation of $\mathcal{I}$.*

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

Now let us assume A is an α-MMS allocation for $\hat{\mathcal{I}} = \texttt{order}(\texttt{normalize}(\texttt{reduce}(\mathcal{I}, \epsilon)))$. By Lemma 2.1, we can obtain an allocation B which is α-MMS for $\texttt{normalize}(\texttt{reduce}(\mathcal{I}, \epsilon))$. Theorem 2.2 implies that B is α-MMS

Copyright © 2024
Copyright for this paper is retained by authors

for $\mathcal{I}' = (N', M', \mathcal{V}') = \texttt{reduce}(\mathcal{I}, \epsilon)$. For all agents $i \in N \setminus N'$, $v'_i(B_i) = v_i(B_i)/\text{MMS}_i(\mathcal{I})$. Therefore,

$$\begin{aligned} v_i(B_i) &= v'_i(B_i)\text{MMS}_i(\mathcal{I}) \\ &\geq \alpha\text{MMS}_i(\mathcal{I}')\text{MMS}_i(\mathcal{I}) && (B \text{ is } \alpha\text{-MMS for } \mathcal{I}') \\ &\geq \alpha(1-4\epsilon)\text{MMS}^n_{v_i}(M). && (\text{MMS}^n_{v'_i}(M) \geq 1-4\epsilon \text{ by Theorem 4.1}) \end{aligned}$$

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

5 $(3/4 + \delta)$-MMS allocation for δ-ONI instances

In this section, we prove that for $\delta \leq 3/956$ there exists a $(3/4+\delta)$-MMS allocation if the input is a δ-ONI instance. First we prove that in any δ-ONI instance $\mathcal{I} = ([n], [m], \mathcal{V})$, $m \geq 2n$.

Observation 5.1. *For any $\delta \leq 1/4$, if $\mathcal{I} = ([n], [m], \mathcal{V})$ is δ-ONI, then $m \geq 2n$.*

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

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

See Figure 1 for a better intuition. Note that by Observation 5.1, $m \geq 2n$ and such bag-initialization is possible.

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

Observation 5.2. *For $\delta \leq 1/4$ and instance $\mathcal{I}$, if $\mathcal{I}$ is δ-ONI, then for all agents $i \in N^2(\mathcal{I})$, $v_i(2n+1) < 1/12+\delta$.*

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

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

In this case we run Algorithm 4. For $k \in [n]$, let B_k and $\hat{B}_k \supseteq B_k$ be the k^{th} bag at the beginning and end of Algorithm 4, respectively.

Lemma 5.1. *Let i be any agent who did not receive any bag by the end of Algorithm 4. For all $k \in [n]$ such that $v_i(B_k) \leq 1$, we have $v_i(\hat{B}_k) < 1 + 4\delta/3$.*

Lemma 5.2. *For $\delta \leq \frac{1}{4}$, given a δ-ONI instance with $|N^1_1| \leq n(\frac{1}{4}-\delta)/(\frac{1}{4}+\frac{\delta}{3})$, all agents $i \in N^1_1$ receive a bag of value at least $(3/4+\delta) \cdot MMS_i$ at the end of Algorithm 4.*

Proof. It suffices to prove that all agents $i \in N^1_1$ receive a bag at the end of Algorithm 4. Towards a contradiction, assume that $i \in N^1_1$ does not receive any bag.

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

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

$$\begin{aligned} v_i(M) &= \sum_{k \in [n]} v_i(\hat{B}_k) = \sum_{B \in S} v_i(B) + \sum_{B \in \bar{S}} v_i(B) \\ &< |N^1_1|\left(1 + \frac{4\delta}{3}\right) + \left(n - |N^1_1|\right)\left(\frac{3}{4} + \delta\right) && \text{(Lemma 5.1 and Claim 1)} \\ &\leq n, && (|N^1_1| \leq n(\tfrac{1}{4}-\delta)/(\tfrac{1}{4}+\tfrac{\delta}{3})) \end{aligned}$$

which is a contradiction since $v_i(M) = n$. Thus, all agents $i \in N^1_1$ receive a bag at the end of Algorithm 4. ∎

Copyright © 2024
Copyright for this paper is retained by authors

Algorithm 4 `approxMMS1`$(\mathcal{I}, \delta)$

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

$B_i \leftarrow \{i, 2n - i + 1\}_{i \in [n]}$
$\mathcal{B} = \cup_{i \in [n]} \{B_i\}$
$\alpha = 3/4 + \delta$
while $\exists i \in N, B \in \mathcal{B}$ s.t. $v_i(B) \geq \alpha$ **do**
  $i \leftarrow$ an arbitrary agent s.t. $v_i(B) \geq \alpha$, priority with agents in N_1^1
  $A_i \leftarrow B$
  $\mathcal{B} \leftarrow \mathcal{B} \setminus \{B\}$
  $N \leftarrow N \setminus \{i\}$
  $M \leftarrow M \setminus B$
end while
$J \leftarrow \cup_{B \in \mathcal{B}} B$
for $B \in \mathcal{B}$ **do**
  while $\nexists i \in N$ s.t. $v_i(B) \geq \alpha$ **do**
    $g \leftarrow$ an arbitrary good in $M \setminus J$
    $B \leftarrow B \cup \{g\}$
    $M \leftarrow M \setminus \{g\}$
  end while
  $i \leftarrow$ an arbitrary agent s.t. $v_i(B) \geq \alpha$, priority with agents in N_1^1
  $A_i \leftarrow B$
  $N \leftarrow N \setminus \{i\}$
  $M \leftarrow M \setminus B$
end for
return $\langle A_1, \ldots, A_n \rangle$

REMARK 1. *The last inequality in the proof of Lemma 5.2 is tight for* $|N_1^1| = n(\frac{1}{4} - \delta)/(\frac{1}{4} + \frac{\delta}{3})$.

LEMMA 5.3. *For* $\delta \leq \frac{1}{4}$*, given a* δ*-ONI instance with* $|N_1^1| \leq n(\frac{1}{4} - \delta)/(\frac{1}{4} + \frac{\delta}{3})$*, all agents* $i \in N_2^1$ *receive a bag of value at least* $(3/4 + \delta) \cdot MMS_i$ *at the end of Algorithm 4.*

Proof. It suffices to prove that all agents $i \in N_2^1$ receive a bag at the end of Algorithm 4. Towards a contradiction, assume that $i \in N_2^1$ does not receive any bag.

CLAIM 2. *For all* $k \in [n]$*,* $v_i(\hat{B}_k) \leq 1$.

Proof. The claim trivially holds if $\hat{B}_k = B_k$. Now assume $B_k \subsetneq \hat{B}_k$. Let g be the last good added to $\hat{B}_k$. We have $v_i(\hat{B}_k \setminus g) < 3/4 + \delta$, otherwise g would not be added to $\hat{B}_k$. Also note that $g \geq 2n + 1$ and hence $v_i(g) \leq v_i(2n + 1) < 1/4 - 5\delta$ by the definition of N_2^1. Therefore, we have

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

Thus, the claim holds. ■

Since agent i did not receive a bag, there exists an unallocated bag with value less than 1 for agent i. Therefore, $v_i(M) = \sum_{k \in [n]} v_i(\hat{B}_k) < n$ which is a contradiction. Thus, all agents $i \in N_2^1$ receive a bag at the end of Algorithm 4. □

5.1.1 Agents in N^2 In this section, we prove that all agents in N^2 also receive a bag at the end of Algorithm 4. For the sake of contradiction, assume that agent $i \in N^2$ does not receive a bag at the end of Algorithm 4. Let $A^+ := \{k \in [n] \mid v_i(B_k) > 1\}$ and $A^- := \{k \in [n] \mid v_i(B_k) < 3/4 + \delta\}$ be the indices of the bags satisfying the

Copyright © 2024
Copyright for this paper is retained by authors

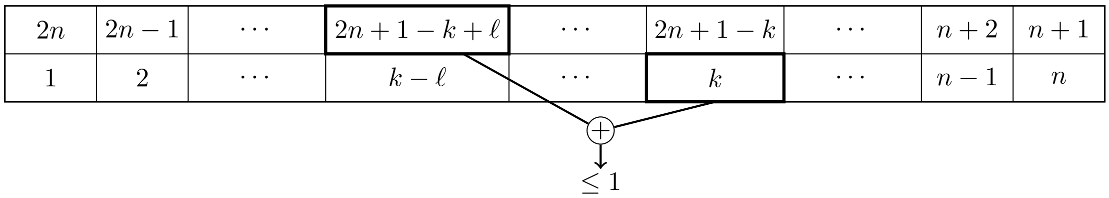

Figure 2: The items $[2n]$ are arranged in a table, where the k^{th} column is $B_k = \{k, 2n+1-k\}$. ℓ is the smallest *shift* such that $v_i(k) + v_i(2n+1-k+\ell) \leq 1$ for all k.

respective constraint. Also, let ℓ be the smallest such that for all $k \in [\ell+1, n]$, $v_i(k) + v_i(2n-k+1+\ell) < 1$. See Figure 2 taken from [3]. [3] proved $\sum_{k \in A^+} v_i(\hat{B}_k) < |A^+| + \ell(\frac{1}{12} + \delta)$.

Lemma 5.4. *[3]* $\sum_{k \in A^+} v_i(\hat{B}_k) < |A^+| + \ell(\frac{1}{12} + \delta)$.

Observation 5.3. *For all* $k \in A^-$, $v_i(\hat{B}_k) < \frac{5}{6} + 2\delta$.

Observation 5.4. *For all* $k \in [n]$, $v_i(B_k) > \frac{1}{2} - 2\delta$.

Observation 5.5. $v_i(M \setminus [2n]) > \ell(\frac{1}{4} - \delta)$.

We are now ready to prove Lemma 5.5.

Lemma 5.5. *For* $\delta \leq 0.011$, *given a* δ*-ONI instance with* $|N_1^1| \leq n(\frac{1}{4} - \delta)/(\frac{1}{4} + \frac{\delta}{3})$, *all agents* $i \in N^2$ *receive a bag of value at least* $(\frac{3}{4} + \delta)$ *at the end of Algorithm 4.*

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

$$v_i(\hat{B}_k) = v_i(B_k) \leq 1. \tag{5.4}$$

We have

$$n = v_i(M) = \sum_{k \in A^-} v_i(\hat{B}_k) + \sum_{k \in A^+} v_i(\hat{B}_k) + \sum_{k \in N \setminus (A^- \cup A^+)} v_i(\hat{B}_k)$$

(Observation 5.3, Lemma 5.4 and Inequality (5.4))

$$< \left(|A^-|(\frac{5}{6} + 2\delta)\right) + \left(|A^+| + \ell(\frac{1}{12} + \delta)\right) + \left(n - |A^-| - |A^+|\right)$$
$$= n - |A^-|(\frac{1}{6} - 2\delta) + \ell(\frac{1}{12} + \delta).$$

Therefore, we have

$$\frac{|A^-|}{\ell} < \frac{1/12 + \delta}{1/6 - 2\delta}. \tag{5.5}$$

Next, we bound the value of the goods in $M \setminus [2n]$ and contradict Inequality (5.5). We have,

$$\begin{aligned}
\ell(\frac{1}{4} - \delta) &\leq v_i(M \setminus [2n]) && \text{(Observation 5.5)} \\
&= \sum_{k \in A^-} \left(v_i(\hat{B}_k) - v_i(B_k)\right) && (M \setminus [2n] = \textstyle\bigcup_{k \in A^-}(\hat{B}_k \setminus B_k)) \\
&< |A^-| \left((\frac{5}{6} + \delta) - (\frac{1}{2} - 2\delta)\right) && \text{(Observation 5.3 and Observation 5.4)} \\
&= |A^-| \cdot (\frac{1}{3} + 3\delta).
\end{aligned}$$

Copyright © 2024
Copyright for this paper is retained by authors

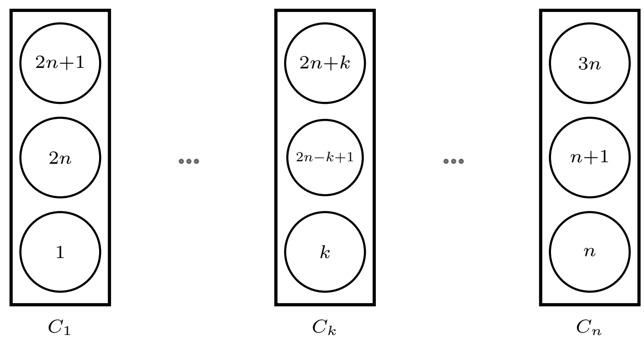

Figure 3: Configuration of Bags $C_1, C_2, \ldots, C_n$

Thus,

$$\frac{|A^-|}{\ell} > \frac{1/4-\delta}{1/3+3\delta} \tag{5.6}$$

Inequalities (5.5) and (5.6) imply that $\frac{1/12+\delta}{1/6-2\delta} > \frac{1/4-\delta}{1/3+3\delta}$, which is a contradiction with $\delta \leq 0.011$. Thus, all agents $i \in N^2$ receive a bag at the end of Algorithm 4. ∎

THEOREM 5.1. *Given any $\delta \leq 0.011$, for all δ-ONI instances where $|N_1^1| \leq n(\frac{1}{4}-\delta)/(\frac{1}{4}+\frac{\delta}{3})$, Algorithm 4 returns a $(\frac{3}{4}+\delta)$-MMS allocation.*

Proof. Since $N = N_1^1 \cup N_2^1 \cup N^2$, by Lemmas 5.2, 5.3 and 5.5 all agents receive a bag of value at least $(\frac{3}{4}+\delta)\cdot \text{MMS}_i$ in Algorithm 4. ∎

5.2 Case 2: $|N_1^1| > n(\frac{1}{4}-\delta)/(\frac{1}{4}+\frac{\delta}{3})$ In this case, we run Algorithm 5. Starting from an ordered normalized $(3/4+\delta)$-irreducible instance, as long as there is a bag B_k with value at least $3/4+\delta$ for some agent, we give B_k to such an agent. The priority is with agents who initially belonged to $N_2^1 \cup N^2$. Therefore, in the remaining instance, all bags are of value less than $3/4+\delta$ for all the remaining agents. We introduce one more reduction rule in this section.

- R_5^α : If $v_i(1)+v_i(2) \geq \alpha$ for some $i \in N$, allocate $\{1,2\}$ to agent i and remove i from N. The priority is with agents in $N_2^1 \cup N^2$.

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

LEMMA 5.6. *For all agents $i \in N_2^1 \cup N^2$ and bags B which is allocated to an agent in $N_2^1 \cup N^2$ during Algorithm 5, $v_i(B) < 3/2+2\delta$.*

LEMMA 5.7. *For $\delta \leq 1/20$, given a δ-ONI instance with $|N_1^1| > n(\frac{1}{4}-\delta)/(\frac{1}{4}+\frac{\delta}{3})$, all agents in $N_2^1 \cup N^2$ receive a bag of value at least $3/4+\delta$ at the end of Algorithm 5.*

Proof. It suffices to prove that all agents $i \in N_2^1 \cup N^2$ receive a bag at the end of Algorithm 5. Towards a contradiction, assume that $i \in N_2^1 \cup N^2$ does not receive any bag.

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

Copyright © 2024
Copyright for this paper is retained by authors

Algorithm 5 `approxMMS2`$(\mathcal{I}, \delta)$

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

```
B_i ← {i, 2n − i + 1}_{i∈[n]}
𝓑 = ∪_{i∈[n]} {B_i}
α = 3/4 + δ
while ∃i ∈ N, B ∈ 𝓑 s.t. v_i(B) ≥ α do
    i ← an arbitrary agent s.t. v_i(B) ≥ α, priority with agents in N_2^1 ∪ N^2
    A_i ← B
    𝓑 ← 𝓑 \ {B}
    N ← N \ {i}
    M ← M \ B
end while
while R_5^α(α) is applicable do
    apply R_5^α(α)
end while
while R_2^α or R_3^α is applicable do
    apply R_k^α for smallest k ∈ {2, 3} s.t. R_k^α is applicable
end while
n ← |N|
C_i ← {i, 2n − i + 1, 2n + i}_{i∈[n]}
for k ← 1 to n do
    while ∄i ∈ N s.t. v_i(C_k) ≥ α do
        g ← an arbitrary good in M \ [3n]
        C_k ← C_k ∪ {g}
        M ← M \ {g}
    end while
    i ← an arbitrary agent s.t v_i(C_k) ≥ α, priority with agents in N_2^1 ∪ N^2
    A_i ← C_k
    N ← N \ {i}
    M ← M \ C_k
end for
return ⟨A_1, ..., A_n⟩
```

CLAIM 3. *For all bags B which is either unallocated or is allocated to an agent in N_1^1, $v_i(B) < 3/4 + \delta$.*

The claim holds since the priority is with agents in $N_2^1 \cup N^2$ and also that we allocate all the bags of value at least $3/4 + \delta$ for some remaining agent.

Let S be the set of bags allocated to agents in $N_2^1 \cup N^2$ and $\bar{S}$ be the set of the remaining bags. We have

$$
\begin{aligned}
n = v_i(M) &= \sum_{B \in S} v_i(B) + \sum_{B \in \bar{S}} v_i(B) \\
&\leq (n - |N_1^1|) \left(\frac{3}{2} + 2\delta\right) + |N_1^1| \left(\frac{3}{4} + \delta\right) && \text{(Lemma 5.6 and Claim 3)} \\
&= \left(\frac{3}{4} + \delta\right) (2n - |N_1^1|) \\
&< n(\frac{3}{4} + \delta)(2 - \frac{\frac{1}{4} - \delta}{\frac{1}{4} + \frac{\delta}{3}}).(|N_1^1| > n(\tfrac{1}{4} - \delta)/(\tfrac{1}{4} + \tfrac{\delta}{3})) \\
&= 3n(\frac{5\delta}{3} + \frac{1}{4})
\end{aligned}
$$

This implies that $\frac{5\delta}{3} + \frac{1}{4} > \frac{1}{3}$. which is a contradiction with $\delta \leq 1/20$. Therefore, all agents $i \in N_2^1 \cup N^2$ receive

Copyright © 2024
Copyright for this paper is retained by authors

a bag at the end of Algorithm 5. ∎

5.2.1 Agents in N_1^1 In this section, we prove that all agents in N_1^1 also receive a bag at the end of Algorithm 5. First, we prove a general lemma that lower bounds the MMS value of an agent after allocating $2k$ goods to k other agents. This way, we can lower bound the MMS value of agents in N_1^1 after the sequence of R_5 rules is applied.

LEMMA 5.8. *Let $i \in N_1^1$ be a remaining agent after no more R_5 is applicable. Then, before applying more reduction rules, $MMS_i \geq 1 - 12\delta$.*

For the sake of contradiction, assume that agent $i \in N_1^1$ does not receive a bag at the end of Algorithm 5. By Lemma 5.8, $\text{MMS}_i \geq 1 - 12\delta$ after applying the sequence of R_5's. By Lemma 2.5, R_2 and R_3 are valid reductions for i and, therefore, $\text{MMS}_i \geq 1 - 12\delta$ at the beginning of the bag-filling phase. Let us abuse the notation and assume the instance at this step is $([n], [m], \mathcal{V})$.

LEMMA 5.9. *If $\delta \leq 1/212$, there exists $k \in [n]$ such that $v_i(C_k) > 1 - 12\delta$.*

Let t be largest s.t. $v_i(C_t) > 1 - 12\delta$ and ℓ be largest such that $v_i(2n + \ell) \geq \delta(26 + 2/3)$.

OBSERVATION 5.6. *If $\delta \leq 3/476$, then $\ell \geq t$.*

LEMMA 5.10. *If $\delta \leq 3/956$, for all $k \leq \min(\ell, n)$, $v_i(C_k) \geq 3/4 + \delta$.*

Note that since i does not receive a bag by the end of Algorithm 5, there must be a remaining bag C_k such that $v_i(C_k) < 3/4 + \delta$. Thus, Lemma 5.10 implies that $\ell < n$ when $\delta \leq 3/956$.

COROLLARY 5.1. (OF LEMMA 5.10) *If $\delta \leq 3/956$, for all $k \leq \ell$, $\hat{C}_k = C_k$.*

OBSERVATION 5.7. $v_i(M \setminus \{1, 2, \ldots, 2n + \ell\}) \geq (n - \ell)(1/4 - 13\delta)$.

LEMMA 5.11. *If $\delta \leq 3/796$, for all $k > \ell$, $v_i(\hat{C}_k \setminus \{k, 2n - k + 1\}) < 1/4 - 13\delta$.*

We are ready to prove Lemma 5.12.

LEMMA 5.12. *For $\delta \leq 3/956$, given a δ-ONI instance with $|N_1^1| > n(\frac{1}{4} - \delta)/(\frac{1}{4} + \frac{\delta}{3})$, all agents in N_1^1 receive a bag of value at least $3/4 + \delta$ at the end of Algorithm 5.*

Proof. It suffices to prove that all agents $i \in N_1^1$ receive a bag at the end of Algorithm 5. Towards a contradiction, assume that $i \in N_1^1$ does not receive any bag. By Lemma 5.9, there exists a $k \in [n]$ such that $v_i(C_k) > 1 - 12\delta$. Recall that ℓ is largest such that $v_i(2n + \ell) \geq \delta(26 + 2/3)$. We have

$$\begin{aligned}
(n - \ell)(\frac{1}{4} - 13\delta) &\leq v_i(M \setminus \{1, 2, \ldots, 2n + \ell\}) && \text{(Observation 5.7)} \\
&= \sum_{k > \ell} v_i(\hat{C}_k \setminus \{k, 2n - k + 1\}) && (\hat{C}_k = C_k \text{ for } k \in [\ell] \text{ by Corollary 5.1}) \\
&< (n - \ell)(\frac{1}{4} - 13\delta), && \text{(Lemma 5.11)}
\end{aligned}$$

which is a contradiction. ∎

THEOREM 5.2. *Given any $\delta \leq 3/956$, for all δ-ONI instances where $|N_1^1| > n(\frac{1}{4} - \delta)/(\frac{1}{4} + \frac{\delta}{3})$, Algorithm 5 returns a $(\frac{3}{4} + \delta)$-MMS allocation.*

Proof. For all other agents i, if $i \in N_2^1 \cup N^2$, by Lemma 5.7, i receives a bag of value at least $\frac{3}{4} + \delta$ and if $i \in N_1^1$, by Lemma 5.12 i receives such a bag. Since $N = N_1^1 \cup N_2^1 \cup N^2$, the theorem follows. ∎

Copyright © 2024
Copyright for this paper is retained by authors

Algorithm 6 $\texttt{mainApproxMMS}(\mathcal{I}, \alpha)$

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

$\epsilon = \alpha - 3/4$
$\delta = 3/956$
$\mathcal{I} = \texttt{order}(\texttt{normalize}(\texttt{reduce}(\mathcal{I}, \epsilon)))$
$N_1^1 = \{i \in [n] \mid \forall j \in [n] : v_i(B_j) \leq 1 \text{ and } v_i(2n+1) \geq 1/4 - 5\delta\}$
if $|N_1^1| \leq n(\frac{1}{4} - \delta)/(\frac{1}{4} + \frac{\delta}{3})$ **then**
  return $\texttt{aprroxMMS1}(\mathcal{I}, \delta)$ ▷ Algorithm 4 in Section 5.1
else
  return $\texttt{aprroxMMS2}(\mathcal{I}, \delta)$ ▷ Algorithm 5 in Section 5.2
end if
return $\langle A_1, \ldots, A_n \rangle$

6 $(3/4 + \epsilon)$-MMS allocations

In this section, we give the complete algorithm $\texttt{mainApproxMMS}(\mathcal{I}, \alpha)$ that achieves an α-MMS allocation for any instance $\mathcal{I}$ with additive valuations and any $\alpha = 3/4 + \epsilon$ for $\epsilon \leq 3/3836$. To this end, first we obtain a δ-ONI instance for $\delta = 4\epsilon/(1 - 4\epsilon)$ by running $\texttt{order}(\texttt{normalize}(\texttt{reduce}(\mathcal{I}, \epsilon)))$. Then depending on whether $|N_1^1| \leq n(\frac{1}{4} - \delta)/(\frac{1}{4} + \frac{\delta}{3})$ or $|N_1^1| > n(\frac{1}{4} - \delta)/(\frac{1}{4} + \frac{\delta}{3})$, we run $\texttt{approxMMS1}$ or $\texttt{approxMMS2}$. The pseudocode of our algorithm $\texttt{mainApproxMMS}(\mathcal{I}, \alpha)$ is shown in Algorithm 6.

Theorem 6.1. *Given any instance $\mathcal{I} = (N, M, \mathcal{V})$ where agents have additive valuations and any $\alpha \leq \frac{3}{4} + \frac{3}{3836}$, $\texttt{mainApproxMMS}(\mathcal{I}, \alpha)$ returns an α-MMS allocation for $\mathcal{I}$.*

Proof. Let $\epsilon = \alpha - 3/4$ and $\hat{\mathcal{I}} = \texttt{order}(\texttt{normalize}(\texttt{reduce}(\mathcal{I}, \epsilon)))$. Then by Theorem 4.2, $\hat{\mathcal{I}}$ is ordered, normalized and $(\frac{3}{4} + \frac{4\epsilon}{1-4\epsilon})$-irreducible ($\frac{4\epsilon}{1-4\epsilon}$-ONI). Since $\epsilon \leq \frac{3}{3836}$, $\frac{4\epsilon}{1-4\epsilon} \leq \frac{3}{956} = \delta$. Thus, $\hat{\mathcal{I}}$ is δ-ONI. Furthermore, from any β-MMS allocation of $\hat{\mathcal{I}}$ one can obtain a $\min(\frac{3}{4} + \epsilon, (1 - 4\epsilon)\beta)$-MMS allocation of $\mathcal{I}$.

By Theorem 5.1, given any $\delta \leq 3/956$, for all δ-ONI instances where $|N_1^1| \leq n(\frac{1}{4} - \delta)/(\frac{1}{4} + \frac{\delta}{3})$, $\texttt{approxMMS1}$ returns a $(\frac{3}{4} + \delta)$-MMS allocation. Also, by Theorem 5.2, for all δ-ONI instances where $|N_1^1| > n(\frac{1}{4} - \delta)/(\frac{1}{4} + \frac{\delta}{3})$, $\texttt{approxMMS2}$ returns a $(\frac{3}{4}+\delta)$-MMS allocation. Therefore, $\texttt{mainApproxMMS}(\mathcal{I}, \alpha)$ returns a $\min(\frac{3}{4}+\epsilon, (1-4\epsilon)(\frac{3}{4}+\delta))$-MMS allocation of $\mathcal{I}$. We have

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

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

References

[1] Aigner-Horev, E., and Segal-Halevi, E. Envy-free matchings in bipartite graphs and their applications to fair division. *Inf. Sci. 587* (2022), 164–187.

[2] Akrami, H., and Garg, J. Breaking the 3/4 barrier for approximate maximin share. *CoRR abs/2307.07304* (2023).

[3] Akrami, H., Garg, J., Sharma, E., and Taki, S. Simplification and improvement of MMS approximation. In *Proc. 32nd Intl. Joint Conf. Artif. Intell. (IJCAI)* (2023).

[4] Akrami, H., Garg, J., and Taki, S. Improving approximation guarantees for maximin share. *CoRR abs/2307.12916* (2023).

[5] Akrami, H., Seddighin, M., Mehlhorn, K., and Shahkarami, G. Randomized and deterministic maximin-share approximations for fractionally subadditive valuations. *CoRR abs/2308.14545* (2023).

Copyright © 2024
Copyright for this paper is retained by authors

[6] Amanatidis, G., Birmpas, G., Christodoulou, G., and Markakis, E. Truthful allocation mechanisms without payments: Characterization and implications on fairness. In *ACM Conference on Economics and Computation (EC)* (2017), pp. 545–562.
[7] Amanatidis, G., Birmpas, G., and Markakis, E. On truthful mechanisms for maximin share allocations. In *International Joint Conference on Artificial Intelligence (IJCAI)* (2016), pp. 31–37.
[8] Amanatidis, G., Markakis, E., Nikzad, A., and Saberi, A. Approximation algorithms for computing maximin share allocations. *ACM Transactions on Algorithms (TALG) 13*, 4 (2017), 1–28.
[9] Aziz, H., Li, B., and Wu, X. Strategyproof and approximately maxmin fair share allocation of chores. In *International Joint Conference on Artificial Intelligence (IJCAI)* (2019), pp. 60–66.
[10] Aziz, H., and Mackenzie, S. A discrete and bounded envy-free cake cutting protocol for any number of agents. In *Proc. 57th Symp. Foundations of Computer Science (FOCS)* (2016), pp. 416–427.
[11] Aziz, H., and Mackenzie, S. A discrete and bounded envy-free cake cutting protocol for four agents. In *Proc. 48th Symp. Theory of Computing (STOC)* (2016), pp. 454–464.
[12] Aziz, H., Rauchecker, G., Schryen, G., and Walsh, T. Algorithms for max-min share fair allocation of indivisible chores. In *AAAI Conference on Artificial Intelligence (AAAI)* (2017).
[13] Babaioff, M., Ezra, T., and Feige, U. Fair-share allocations for agents with arbitrary entitlements. In *ACM Conference on Economics and Computation (EC)* (2021), pp. 127–127.
[14] Babaioff, M., Ezra, T., and Feige, U. On best-of-both-worlds fair-share allocations. In *Proc. 18th Conf. Web and Internet Economics (WINE)* (2022), vol. 13778, pp. 237–255.
[15] Babaioff, M., and Feige, U. Fair shares: Feasibility, domination and incentives. In *ACM Conference on Economics and Computation (EC)* (2022).
[16] Bansal, N., and Sviridenko, M. The Santa Claus problem. In *Symp. Theory of Computing (STOC)* (2006), pp. 31–40.
[17] Barman, S., Biswas, A., Krishnamurthy, S., and Narahari, Y. Groupwise maximin fair allocation of indivisible goods. In *AAAI Conference on Artificial Intelligence (AAAI)* (2018).
[18] Barman, S., Ghalme, G., Jain, S., Kulkarni, P., and Narang, S. Fair division of indivisible goods among strategic agents. In *International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)* (2019), p. 1811–1813.
[19] Barman, S., and Krishnamurthy, S. K. Approximation algorithms for maximin fair division. *ACM Transactions on Economics and Computation (TEAC) 8*, 1 (2020), 1–28.
[20] Bei, X., Igarashi, A., Lu, X., and Suksompong, W. The price of connectivity in fair division. *SIAM Journal on Discrete Mathematics 36*, 2 (2022), 1156–1186.
[21] Biswas, A., and Barman, S. Fair division under cardinality constraints. In *International Joint Conference on Artificial Intelligence (IJCAI)* (2018), pp. 91–97.
[22] Bouveret, S., and Lemaître, M. Characterizing conflicts in fair division of indivisible goods using a scale of criteria. *Autonomous Agents and Multi-Agent Systems 30*, 2 (2016), 259–290.
[23] Budish, E. The combinatorial assignment problem: Approximate competitive equilibrium from equal incomes. *Journal of Political Economy 119*, 6 (2011), 1061–1103.
[24] Caragiannis, I., Kurokawa, D., Moulin, H., Procaccia, A. D., Shah, N., and Wang, J. The unreasonable fairness of maximum Nash welfare. *ACM Trans. Economics and Comput. 7*, 3 (2019), 12:1–12:32.
[25] Chakrabarty, D., Chuzhoy, J., and Khanna, S. On allocating goods to maximize fairness. In *Proc. 50th Symp. Foundations of Computer Science (FOCS)* (2009), pp. 107–116.
[26] Chaudhury, B. R., Kavitha, T., Mehlhorn, K., and Sgouritsa, A. A little charity guarantees almost envy-freeness. *SIAM Journal on Computing 50*, 4 (2021), 1336–1358.
[27] Farhadi, A., Ghodsi, M., Hajiaghayi, M. T., Lahaie, S., Pennock, D., Seddighin, M., Seddighin, S., and Yami, H. Fair allocation of indivisible goods to asymmetric agents. *Journal of Artificial Intelligence Research 64* (2019), 1–20.
[28] Feige, U., and Norkin, A. Improved maximin fair allocation of indivisible items to three agents. *arXiv abs/2205.05363* (2022).
[29] Feige, U., Sapir, A., and Tauber, L. A tight negative example for MMS fair allocations. In *International Conference on Web and Internet Economics (WINE)* (2021), pp. 355–372.
[30] Foley, D. Resource allocation and the public sector. *Yale Econ Essays 7*, 1 (1967), 45–98.
[31] Garg, J., McGlaughlin, P., and Taki, S. Approximating maximin share allocations. In *Symposium on Simplicity in Algorithms (SOSA)* (2019), pp. 20:1–20:11.
[32] Garg, J., and Taki, S. An improved approximation algorithm for maximin shares. *Artificial Intelligence* (2021).
[33] Gates, V., Griffiths, T. L., and Dragan, A. D. How to be helpful to multiple people at once. *Cognitive Science 44* (2020).
[34] Ghodsi, M., HajiAghayi, M., Seddighin, M., Seddighin, S., and Yami, H. Fair allocation of indivisible goods:

Copyright © 2024
Copyright for this paper is retained by authors

Improvements and generalizations. In *Proc. 19th Conf. Economics and Computation (EC)* (2018), pp. 539–556. (arXiv:1704.00222).

[35] Gourvès, L., and Monnot, J. On maximin share allocations in matroids. *Theoretical Computer Science 754* (2019), 50–64.

[36] Hosseini, H., and Searns, A. Guaranteeing maximin shares: Some agents left behind. In *International Joint Conference on Artificial Intelligence (IJCAI)* (2021), pp. 238–244.

[37] Hosseini, H., Searns, A., and Segal-Halevi, E. Ordinal maximin share approximation for goods. *J. Artif. Intell. Res. 74* (2021).

[38] Huang, X., and Lu, P. An algorithmic framework for approximating maximin share allocation of chores. In *ACM Conference on Economics and Computation (EC)* (2021), pp. 630–631.

[39] Huang, X., and Segal-Halevi, E. A reduction from chores allocation to job scheduling. *arXiv abs/2302.04581* (2023).

[40] Kurokawa, D., Procaccia, A. D., and Wang, J. Fair enough: Guaranteeing approximate maximin shares. *Journal of the ACM (JACM) 65*, 2 (2018), 1–27.

[41] Li, Z., and Vetta, A. The fair division of hereditary set systems. *ACM Transactions on Economics and Computation (TEAC) 9*, 2 (2021), 1–19.

[42] Lipton, R. J., Markakis, E., Mossel, E., and Saberi, A. On approximately fair allocations of indivisible goods. In *Proceedings 5th ACM Conference on Electronic Commerce (EC)* (2004), pp. 125–131.

[43] Procaccia, A. D., and Wang, J. Fair enough: Guaranteeing approximate maximin shares. In *Proc. 15th Conf. Economics and Computation (EC)* (2014), pp. 675–692.

[44] Steinhaus, H. The problem of fair division. *Econometrica 16* (1948), 101–104.

[45] Truszczynski, M., and Lonc, Z. Maximin share allocations on cycles. *Journal of Artificial Intelligence Research 69* (2020), 613–655.

[46] Uziahu, G. B., and Feige, U. On fair allocation of indivisible goods to submodular agents. *arXiv abs/2303.12444* (2023).

[47] Varian, H. Equity, envy and efficiency. *J. Econom. Theory 29*, 2 (1974), 217–244.

[48] Woeginger, G. J. A polynomial-time approximation scheme for maximizing the minimum machine completion time. *Operations Research Letters 20*, 4 (1997), 149–154.

[49] Xiao, M., Qiu, G., and Huang, S. MMS allocations of chores with connectivity constraints: New methods and new results. In *Proceedings of the 2023 International Conference on Autonomous Agents and Multiagent Systems, AAMAS 2023, London, United Kingdom, 29 May 2023 - 2 June 2023* (2023), N. Agmon, B. An, A. Ricci, and W. Yeoh, Eds., ACM, pp. 2886–2888.

Copyright © 2024
Copyright for this paper is retained by authors