Multiple Birds with One Stone: Beating 1/2 for EFX and GMMS via Envy Cycle Elimination

Georgios Amanatidis[a,*], Evangelos Markakis[b], Apostolos Ntokos[b]

[a]*Sapienza University of Rome, Department of Computer, Control and Management Engineering*
Via Ariosto 25, 00185 Rome, Italy
[b]*Athens University of Economics and Business, Department of Informatics*
Patision 76, Athens 104 34, Greece

Abstract

Several relaxations of envy-freeness, tailored to fair division in settings with indivisible goods, have been introduced within the last decade. Due to the lack of general existence results for most of these concepts, great attention has been paid to establishing approximation guarantees. In this work, we propose a simple algorithm that is *universally fair* in the sense that it returns allocations that have good approximation guarantees with respect to four such fairness notions at once. In particular, this is the first algorithm achieving a $(\phi - 1)$-approximation of *envy-freeness up to any good* (EFX) and a $\frac{2}{\phi+2}$-approximation of *groupwise maximin share fairness* (GMMS), where ϕ is the golden ratio ($\phi \approx 1.618$). The best known approximation factor, in polynomial time, for either one of these fairness notions prior to this work was 1/2. Moreover, the returned allocation achieves *envy-freeness up to one good* (EF1) and a 2/3-approximation of *pairwise maximin share fairness* (PMMS). While EFX is our primary focus, we also exhibit how to fine-tune our algorithm and further improve the guarantees for GMMS or PMMS.

Finally, we show that GMMS—and thus PMMS and EFX—allocations always exist when the number of goods does not exceed the number of agents by more than two.

Keywords: fair division, envy-freeness up to any good, groupwise maximin share fairness, envy cycle elimination

1. Introduction

The mathematical study of *fair division* has a long and intriguing history, starting with the formal introduction of the cake-cutting problem by Banach, Knaster and Steinhaus [33]. Ever since, we have seen the emergence of several fairness criteria, such as the classic notion of *envy-freeness*, that has a dominant role in the literature, see e.g., Brams and Taylor [10], Moulin [30], Robertson and Webb [32], Brandt et al. [11] and references therein.

On the other hand, the computational study of finding fair allocations when the resources are *indivisible items* is more recent. It is motivated by the realization that envy-freeness and other classic fairness notions are too demanding for the discrete setting. In particular, even with two agents and one item, it is impossible to produce an allocation with *any* reasonable worst-case approximation guarantee with respect to envy-freeness.

Within the last decade, these considerations have led to natural relaxations of envy-freeness, which are more suitable for the context of indivisible goods. The most prominent examples, that are also the focus of our work, include the notions of envy-freeness up to one good (EF1) and up to any good (EFX), maximin share fairness (MMS), as well as pairwise and groupwise maximin share fairness (PMMS and GMMS respectively). These relatively new concepts breathed new life into the field of fair division, but they do not come without their issues. Most importantly, although they are generally easier to satisfy than envy-freeness, proving existence results has turned out to be a very challenging task—with the exception of EF1. For instance, it is an open problem to resolve whether PMMS and EFX allocations always exist, even for three and four agents with additive valuation functions, respectively. Surprisingly, existence remains unresolved even when the number of items is just slightly larger than the number of agents.

A reasonable approach is to focus on approximate versions of these relaxations. Indeed, this has led to a series of positive results, obtaining constant factor approximation algorithms for all the aforementioned relaxed criteria (see Related Work). However, improving on the currently known factors seems to be approaching a stagnation point. For example, soon after the introduction of EFX, a 1/2-approximation was established [31], but there has been no progress beyond 1/2, despite the active interest on this notion.

We should also stress that these notions capture quite different aspects of fairness. A good approximation of any one of EF1, EFX, MMS and PMMS does not necessarily imply particularly strong guarantees for any of the others [5]. Hence, it becomes compelling to ask for allocations that attain good guarantees with respect to several fairness notions simultaneously. Such results are rather scarce in the literature, e.g., [8, 20] or are purely existential [14].

Motivated by the lack of such *universally fair* algorithms, we look at the problem of computing allocations that (approximately) satisfy several fairness notions at the same time. Along the way, we aim to improve the state-of-the-art for two of these notions, namely EFX and GMMS. Somewhat unexpectedly, to do so we rely on simple subroutines that have been repeatedly used in fair division before.

Contribution. Our main contribution is an algorithm that is universally fair, in the sense that it achieves a better than 1/2-approximation for all the notions under consideration. The main results can be summarized in the following statement.

Main Theorem. *We can efficiently compute an allocation that is simultaneously*

   i) EFX *up to a factor of* 0.618,

   ii) GMMS *up to a factor of* 0.553 *(thus, ditto for* MMS*),*

   iii) EF1, *and*

   iv) PMMS *up to a factor of* 0.667.

We view parts i) and ii) of breaking the 1/2-approximation barrier for EFX and GMMS, as the highlights of this work. These desirable properties are attained by Algorithm 3

(Section 3). We also suggest variations with improved guarantees for one notion at the expense of the others. The factors achieved by Algorithm 3 and its variants, compared against the state of the art for each notion, are shown in Table 1.

	EFX	EF1	GMMS	PMMS
Best known (poly-time)	0.5	1	0.5	0.781
Algorithm 3	0.618	1	0.553	0.667
Variant in Thm. 4.7	0.6	1	0.571	0.667
Variant in Thm. 4.10	0.618	0.894	0.553	0.717

Table 1: Summary of our results and state of the art. Known results in the first row are due to Plaut and Roughgarden [31], Lipton et al. [27], Barman et al. [8], and Kurokawa [25], respectively.

At a technical level, our results are making use of two algorithms that are known to produce only EF1 allocations. The first one is a simple draft algorithm and the second one is the envy-cycle-elimination algorithm of Lipton et al. [27]. Although these algorithms on their own do not possess *any* good approximations with respect to EFX or GMMS, our main insight is that by carefully combining parametric versions of these algorithms, we can obtain approximation guarantees for all the fairness criteria of interest here.

In Section 5, we return to the intriguing issue of existence. We show that GMMS—and thus PMMS and EFX—allocations always exist, and can be found efficiently, when the number of goods does not exceed the number of agents by more than two. While this is a simple case, it is still non-trivial to tackle and has remained unresolved. Quite surprisingly, the idea of envy cycle elimination again comes to the rescue, after we carefully alter a small part of the instance.

Related work. Envy-freeness was initially suggested by Gamow and Stern [19], and more formally by Foley [18] and Varian [35]. Regarding the relaxations of envy-freeness, EF1 was defined by Budish [12], but it was also implicit in the work of Lipton et al. [27]. Budish also defined the notion of maximin shares, based on concepts by Moulin [29]. Later on, Caragiannis et al. [14] introduced the notions of EFX and PMMS, and even more recently, Barman et al. [8] proposed to study GMMS allocations. Further variants and generalizations of the criteria we present here have also been considered, see e.g., Suksompong [34].

EF1 allocations are known to be efficiently computable by the envy-cycle-elimination algorithm of Lipton et al. [27]. For all other notions, the focus has been on approximation algorithms since existence is either not guaranteed or is still an open problem. The most well studied notion is MMS with a series of positive results [4, 26, 7, 22], and best known approximation of 3/4 [24, 21]. Exact and approximate EFX allocations with both additive and general valuations were studied by Plaut and Roughgarden [31], achieving the currently best 1/2-approximation. Recently, a polynomial time algorithm with the same guarantee has been obtained by Chan et al. [15]. The same factor is also the best known for GMMS allocations in polynomial time by Barman et al. [8] via a variant of envy cycle elimination. Finally, the currently best approximation of 0.781 for PMMS is due to Kurokawa [25], using an approach similar to ours. Connections between the approximate versions of these criteria have been investigated by Amanatidis et al. [5]; see Appendix B for a comparison of these implications with our results.

3

Some of these fairness criteria have also been studied in combination with other objectives, such as Pareto optimality [9], truthfulness [2, 3] or maximizing the Nash welfare [14, 13, 17].

In a very recent manuscript, Chaudhury et al. [16] show that EFX allocations always exist for instances with three agents and additive valuation functions. While the existence of EFX allocations remains an open problem for more than three additive agents, it is possible to compute such allocations when it is allowed to discard some of the goods [13, 17] or when the valuation functions are special cases of additive functions [1, 6]. In [17], in parallel and independently of our work, Chaudhury et al. [17] also improve the $1/2$ factor for GMMS but not in polynomial time. The focus of their work is different and involves algorithms for EFX allocations by discarding a relatively small number of items. As an implication of their main results, they too follow closely the proof of Proposition 3.4 of [5], albeit from a different starting point, and obtain a pseudo-polynomial time $4/7$-approximation algorithm. This matches the guarantee of our Theorem 4.7 for GMMS. However, to the best of our knowledge, this algorithm does not directly provide any good approximation for EFX.

2. Preliminaries

Let $N = \{1, 2, \ldots, n\}$ be a set of n agents and M be a set of m indivisible items. Unless otherwise stated, we assume that each agent is associated with a monotone, *additive* valuation function, i.e., for $S \subseteq M$, $v_i(S) = \sum_{g \in S} v_i(\{g\})$. For simplicity, we write $v_i(g)$ instead of $v_i(\{g\})$, for $g \in M$. Monotonicity in this additive setting is equivalent to all items being *goods*, i.e., $v_i(g) \geq 0$ for every $i \in N, g \in M$. For the algorithms presented in this work, we assume that their input contains the valuation function of each involved agent, i.e., $v_i(g)$ is given to the algorithm for every agent i and good g.

We consider the most standard setting in fair division, where we want to allocate all the goods to the agents (*no free disposal*). An allocation of M to the n agents is therefore a partition, $\mathcal{A} = (A_1, \ldots, A_n)$, where $A_i \cap A_j = \emptyset$ and $\cup_i A_i = M$. By $\Pi_n(M)$ we denote the set of all partitions of a set M into n bundles.

Although we allow for multiple goods to have the exact same value for a specific agent, we assume a deterministic tie-breaking rule for the goods (e.g., break ties lexicographically). This way we may abuse the notation and write $g = \arg\max_{h \in M} v_i(h)$ instead of "let g be the lexicographically first element of $\arg\max_{h \in M} v_i(h)$".

2.1. Fairness Concepts

All the fairness notions we work with are relaxations of the classic notion of envy-freeness.

Definition 2.1. *An allocation $\mathcal{A} = (A_1, \ldots, A_n)$ is envy-free (EF), if for every $i, j \in N$, $v_i(A_i) \geq v_i(A_j)$.*

As envy-freeness is too strong to ask for, when we deal with indivisible goods, several relaxed fairness notions have been introduced so as to obtain meaningful positive results. We start with two additive relaxations, and their approximate versions, where an agent may envy another agent, but only by an amount dependent on the value of a single good in the other agent's bundle.

Definition 2.2. *An allocation $\mathcal{A} = (A_1, \ldots, A_n)$ is an*

a) *α-EF1 allocation (α-envy-free up to one good), if for every pair of agents $i, j \in N$, with $A_j \neq \emptyset$, there exists a good $g \in A_j$, such that $v_i(A_i) \geq \alpha \cdot v_i(A_j \setminus \{g\})$.*

b) *α-EFX allocation (α-envy-free up to any good), if for every pair $i, j \in N$, with $A_j \neq \emptyset$ and every good $g \in A_j$, it holds that $v_i(A_i) \geq \alpha \cdot v_i(A_j \setminus \{g\})$.[1]*

Of course, for $\alpha = 1$ we obtain precisely the notions of envy-freeness up to one good (EF1) [12] and envy-freeness up to any good (EFX) [14]. It is easy to see that EF implies EFX, which in turn implies EF1.

On a different direction, an interesting family of fairness criteria has been developed around the notion of *maximin shares*, also proposed by Budish [12]. The idea behind maximin shares is to capture the worst-case guarantees of generalizing the famous cut-and-choose protocol to multiple agents: Suppose agent i is asked to partition the goods into n bundles, while knowing that the other agents will choose a bundle before her. In the worst case, she will be left with her least valuable bundle. Assuming that agents are risk-averse, agent i would choose a partition that maximizes the minimum value of a bundle. This gives rise to the following definition.

Definition 2.3. *Given n agents, and a subset $S \subseteq M$ of goods, the n-maximin share of agent i with respect to S is:*

$$\boldsymbol{\mu}_i(n, S) = \max_{\mathcal{A} \in \Pi_n(S)} \min_{A_j \in \mathcal{A}} v_i(A_j).$$

From the definition, it directly follows that $n \cdot \boldsymbol{\mu}_i(n, S) \leq v_i(S)$. When $S = M$, this quantity is just called the *maximin share* of agent i. We say that $\mathcal{T} \in \Pi_n(M)$ is an *n-maximin share defining partition* for agent i, if $\min_{T_j \in \mathcal{T}} v_i(T_j) = \boldsymbol{\mu}_i(n, M)$. When it is clear from context what n and M are, we simply write $\boldsymbol{\mu}_i$ instead of $\boldsymbol{\mu}_i(n, M)$.

The most popular fairness notion based on maximin shares, referred to as *maximin share fairness*, asks for a partition that gives each agent her (approximate) maximin share.

Definition 2.4. *An allocation $\mathcal{A} = (A_1, \ldots, A_n)$ is called an α-MMS (α-maximin share) allocation if $v_i(A_i) \geq \alpha \cdot \boldsymbol{\mu}_i$, for every $i \in N$.*

Variations of maximin share fairness have also been proposed. Here we focus on two notable examples. The first one, *pairwise maximin share fairness*, is related but not directly comparable to MMS and was introduced by Caragiannis et al. [14]. The idea is to demand an MMS-type guarantee but for any *pair* of agents. That is, we can think of an agent i as considering the combined bundle of herself and another agent and requesting to receive at least her maximin share of this bundle if split into two subsets.

Definition 2.5. *An allocation $\mathcal{A} = (A_1, \ldots, A_n)$ is called an α-PMMS (α-pairwise maximin share) allocation if for every pair of agents $i, j \in N$, $v_i(A_i) \geq \alpha \cdot \boldsymbol{\mu}_i(2, A_i \cup A_j)$.*

---

[1] The original definition required the condition to hold for all $g \in A_j$ with $v_i(g) > 0$. This is often dropped in the literature, under the assumption that all values are positive [31, 13]. For our work *neither* assumption is needed.

Taking this one step further, we can demand an allocation to have an MMS-type guarantee for *any subset* of agents. This is referred to as *groupwise maximin share fairness*, introduced by Barman et al. [8].

Definition 2.6. *An allocation $\mathcal{A} = (A_1, \ldots, A_n)$ is called an α-GMMS (α-groupwise maximin share) allocation if for every subset of agents $N' \subseteq N$ and any agent $i \in N'$,*
$$v_i(A_i) \geq \alpha \cdot \boldsymbol{\mu}_i(|N'|, \cup_{j \in N'} A_j).$$

In Definitions 2.4, 2.5, and 2.6, when $\alpha = 1$, we refer to the corresponding allocations as MMS, PMMS, and GMMS allocations respectively. It is clear that the notion of GMMS is stronger than both MMS and PMMS. Further, it has been observed that EF is stronger than GMMS [8] and, when all values are positive, PMMS is stronger than EFX [14]. It should be noted, however, that the approximate versions of these notions are related in non-straightforward ways [5] (see also Appendix B).

An example illustrating the different fairness criteria can be found in Appendix A.

2.2. Known EF1 Algorithms

Among the fairness notions defined above, EF1 is the only one for which we know that it can always be achieved. Furthermore, two simple algorithms are already known for computing such allocations in polynomial time. We state below a parametric version of these algorithms so that they can run for a limited number of steps or on a strict subset of the goods, as we are going to use them later as subroutines.

In order to define the *envy-cycle-elimination algorithm* (Algorithm 1) of Lipton et al. [27], we first need to introduce the notion of an *envy graph*. Suppose we have a partial allocation $\mathcal{P} = (P_1, \ldots, P_n)$, i.e., an allocation of a strict subset of M. We define the directed envy graph $G_{\mathcal{P}} = (N, E_{\mathcal{P}})$, where $(i, j) \in E_{\mathcal{P}}$ if and only if agent i currently envies agent j, i.e., $v_i(P_i) < v_i(P_j)$. Algorithm 1 builds an allocation one good at a time; in each step, an agent that no one envies receives the next available good. To ensure that such an agent always exists, the algorithm identifies cycles that are created in the envy graph and eliminates them by appropriately reallocating some of the current bundles.

Regarding tie-breaking in line 10 of the algorithm, we assume that agent i is the lexicographically first node of $G_{\mathcal{P}}$ with in-degree 0. Below we summarize the main known properties of Algorithm 1 that we will utilize in our analysis.

Theorem 2.7 (Follows by Lipton et al. [27]). *Let $\mathcal{P}$ be any EF1 partial allocation and $M' = M \setminus \cup_{i=1}^n P_i$. Then,*

a) *at the end of each iteration of the* for *loop, the resulting partial allocation is EF1. Hence, the algorithm terminates with an EF1 allocation in polynomial time. This holds even for agents with general monotone valuation functions.*

b) *Fix an agent i, and let A_i be the bundle assigned to i at the end of some iteration of the* for *loop. If A_i' is assigned to i at the end of a future iteration, then $v_i(A_i') \geq v_i(A_i)$.*

The first property of Theorem 2.7 simply says that the EF1 property is maintained during the execution of the algorithm, given an initial EF1 allocation. The second property states that agents only get happier throughout the course of the algorithm, since they keep getting better and better bundles.

For additive valuation functions there is, in fact, an even simpler greedy algorithm, referred to in the literature as the *round-robin algorithm*, or the *draft algorithm* (Algorithm

---

Algorithm 1: Envy-Cycle-Elimination($N, \mathcal{P}, M'$)

where N: set of agents, $\mathcal{P}$: initial partial allocation, M': set of unallocated goods

---

1 Construct the envy graph $G_{\mathcal{P}}$
2 **for** *every $g \in M'$ in lexicographic order* **do**
3   **while** *there is no node of in-degree 0 in $G_{\mathcal{P}}$* **do**
4   Find a cycle $j_1 \rightarrow j_2 \rightarrow \ldots \rightarrow j_r \rightarrow j_1$ in $G_{\mathcal{P}}$
5 $B = P_{j_1}$
6   **for** $k = 1$ *to* $r - 1$ **do**
7 $P_{j_k} = P_{j_{k+1}}$ /* shift the bundles */
8 $P_{j_r} = B$
9   Update $G_{\mathcal{P}}$
10   Let $i \in N$ be a node of in-degree 0
11 $P_i = P_i \cup \{g\}$
12   Update $G_{\mathcal{P}}$
13 **return** $\mathcal{P}$

---

2) that also outputs EF1 allocations, see e.g., [28]. Given a fixed ordering of the agents, they simply pick their favourite unallocated good one by one, according to that ordering, until there are no goods left.

Algorithm 2: Round-Robin($N, \mathcal{P}, M', \ell, \tau$)

where N: set of agents, $\mathcal{P}$: partial allocation, M': set of unallocated goods, ℓ: an ordering of N, τ: number of steps

---

1 $k = 1$
2 **while** $M' \neq \emptyset$ *and* $\tau > 0$ **do**
3 $g = \arg\max_{h \in M'} v_{\ell[k]}(h)$
4 $P_{\ell[k]} = P_{\ell[k]} \cup \{g\}$
5 $M' = M' \setminus \{g\}$
6 $k = k + 1 \mod n$
7 $\tau = \tau - 1$
8 **return** $(\mathcal{P}, M')$

Theorem 2.8. *Let ℓ be any ordering of N and $\mathcal{P}_{\emptyset} = (\emptyset, \ldots, \emptyset)$. Then Algorithm 2 with input $(N, \mathcal{P}_{\emptyset}, M, \ell, |M|)$ produces an EF1 allocation in polynomial time.*

3. A Simple Universally Fair Algorithm

As mentioned above, Algorithm 3 is built on Algorithms 1 and 2. In particular, it first runs a simple preprocessing step (Algorithm 4) that determines an appropriate ordering ℓ of the set of agents N. Then, it suffices to run only two rounds of the round-robin algorithm, once with respect to ℓ and once with respect to the reverse of ℓ (the second run is also restricted to a subset of the agents), and finally run the envy-cycle-elimination

algorithm on the remaining instance. It should be noted here that the preprocessing step is mostly introduced to facilitate the presentation and the analysis of the algorithm. As it can be seen by its description, Algorithm 4 could be combined with the first run of the round-robin algorithm. Indeed, the final assignments for the h_is in Algorithm 4 are exactly the goods that the agents receive in line 3 of Algorithm 3 (see also Lemma 4.1 in the next section).

Algorithm 3: Draft-and-Eliminate(N, M)

1 $(\ell, n') = \text{Preprocessing}(N, M)$
2 Let $\mathcal{A} = (A_1, \ldots, A_n)$ with $A_i = \emptyset$ for each $i \in N$
3 $(\mathcal{A}, M') = \text{Round-Robin}(N, \mathcal{A}, M, \ell, n)$
4 $\ell^R = (\ell[n], \ell[n-1], \ldots, \ell[1])$
5 $(\mathcal{A}, M') = \text{Round-Robin}(N, \mathcal{A}, M', \ell^R, n - n')$
6 $\mathcal{A} = \text{Envy-Cycle-Elimination}(N, \mathcal{A}, M')$
7 return $\mathcal{A}$

As it is customary, we use ϕ to denote the *golden ratio*. That is, ϕ is the positive solution to the quadratic equation $x^2 - x - 1 = 0$. Recall that $\phi = \frac{1+\sqrt{5}}{2} \approx 1.618$ and that $\phi - 1 = \phi^{-1} \approx 0.618$.

Algorithm 4: Preprocessing(N, M)

1 $L = \emptyset; \ \ A = N; \ \ k = 1$
2 while $A \neq \emptyset$ **do**
3 $\quad$ Let i be the lexicographically first agent of A
4 $\quad$ $h_i = \arg\max_{g \in M} v_i(g)$
5 $\quad$ $t_i = m - |M| + 1$ $\hspace{4cm}$ /* i's `timestamp` */
6 $\quad$ Let $R = (N \setminus (A \cup L)) \cup \{i\}$
7 $\quad$ $j = \arg\max_{t \in R} v_i(h_t)$
8 $\quad$ **if** $\phi \cdot v_i(h_i) < v_i(h_j)$ **then**
9 $\quad\quad$ $h_i = h_j$
10 $\quad\quad$ $L = L \cup \{i\}$
11 $\quad\quad$ $\ell[k] = i$
12 $\quad\quad$ $k = k + 1$
13 $\quad\quad$ $A = (A \setminus \{i\}) \cup \{j\}$
14 $\quad$ **else**
15 $\quad\quad$ $A = A \setminus \{i\}$
16 $\quad\quad$ $M = M \setminus \{h_i\}$
17 for *every $i \in N \setminus L$ in order of increasing timestamp t_i* **do**
18 $\quad$ $\ell[k] = i$
19 $\quad$ $k = k + 1$
20 return $(\ell, |L|)$

Before we move to the analysis of our algorithm, it is useful to build some more

intuition on how things work. The preprocessing part essentially reorders N so that
the first few agents (namely, the first $|L|$ agents) are *quite happy* with their pick in
the first round of the round-robin subroutine. For the remaining agents, we make sure
that they get a second good before we move to the envy-cycle-elimination algorithm.
To do so in a "balanced" way, these agents pick goods in reverse order. The resulting
partial allocation, where everyone receives one or two goods, turns out to have all the
fairness properties we want to achieve at the end, e.g., it is $(\phi - 1)$-EFX with respect
to the currently allocated goods. Crucially, we show that starting from there and then
applying the envy-cycle-elimination algorithm on the remaining instance maintains these
properties.

Coming back to the preprocessing part, the intuition is to simulate a first round of
Algorithm 2 and correct any occurrences of extreme envy. In particular, if an agent envies
someone that chose before her by a factor greater than ϕ, then she is moved to a position
of high priority in the ordering that is created. The agents moved to the first positions
during this process (i.e., agents in L) are guaranteed a good of high value in line 3 of
Algorithm 3. To counterbalance their advantage, they are not allowed to pick a second
good later in line 5.

To see that Algorithm 3 runs in polynomial time, given the properties we have seen
for Algorithms 1 and 2, it suffices to check that the preprocessing step can be efficiently
implemented. Indeed, the *if* branch of the while loop in Algorithm 4 may be executed at
most n times, since agents are irrevocably added to L. Similarly, the *else* branch may
be executed at most n times, as each time the set A becomes smaller and its size never
increases in the other parts of the algorithm.

4. Fairness Guarantees of Algorithm 3

We begin our analysis with two useful lemmata about Algorithm 4. We stress that
within Algorithm 4, every agent i is associated with a distinct good h_i, although nothing
is allocated at this step. The first lemma establishes some useful inequalities regarding
the goods associated with the agents. The second lemma states that this association
actually coincides with the partial allocation produced in line 3 of Algorithm 3.

Recall that the set L, defined in Algorithm 4, contains the agents that get to pick first
in line 3 of Algorithm 3 at the expense of not choosing a second good in line 5. In terms
of Algorithm 3, $L = \{\ell[1], \ell[2], \ldots, \ell[n']\}$). The partition of N into L and $N \setminus L$ is pivotal
for distinguishing the different cases that are relevant in the analysis.

Lemma 4.1. *At the end of the execution of Algorithm 4 with input (N, M), each agent i
is associated with a single good h_i, so that*

    a) $v_i(h_i) > \phi \cdot v_i(g)$, *for any $i \in L$ and $g \in M \setminus \cup_{k=1}^n \{h_k\}$,*

    b) $\phi \cdot v_i(h_i) \geq v_i(h_j)$, *for any $i, j \in N \setminus L$.*

Proof. a) Fix some $i \in L$ and consider the last iteration of the *while* loop where i was the
lexicographically first agent of A, during which i was added to L. We make the distinction
between the initial and the final good associated with i during this iteration by using h_i^{old}
and h_i respectively to denote them. So, initially, i was associated with h_i^{old} which was her
favorite good at the time, among the available ones. Since i was eventually added to L,
we know that the condition in line 8 was true. That is, i's favorite good among the ones

9

associated to an agent not in L, say h_j, was more than ϕ times more valuable than h_i^{old}. By the choice of h_i^{old}, we have $v_i(h_j) > \phi \cdot v_i(h_i^{\text{old}}) \geq \phi \cdot v_i(g)$ for any good g that was not associated to an agent at the time. Note that the set of unassociated goods during the execution of the algorithm only shrinks, and thus the last inequality also holds for any good g that was not associated to any agent till the end. Finally, recall that $h_i = h_j$, as imposed by line 9, to conclude that $v_i(h_i) > \phi \cdot v_i(g)$ for any $g \in M \setminus \cup_{k=1}^n \{h_k\}$.

b) Fix some $i, j \in N \setminus L$. Note that both i and j may be considered multiple times during Algorithm 4, as they may be removed and then added back to the set A several times. We consider two cases, based on the *last* time that each agent was considered (i.e., the last time each of them was the lexicographically first agent in A). If the last time that i was considered by Algorithm 4 happened before the last time that j was considered (i.e., $t_i < t_j$ at the end), the desired inequality is straightforward, as agent i is associated with her favorite good among the available goods, h_i, before agent j, i.e., $v_i(h_i) \geq v_i(h_j)$. On the other hand, if the last time i was considered by Algorithm 4 takes place after the last time that j was considered (i.e., $t_i > t_j$ at the end), suppose that $\phi \cdot v_i(h_i) < v_i(h_j)$. Then, during the last iteration that i was considered, line 8 would be true and i would be (irrevocably) added to L in line 10, which contradicts the choice of i. $\square$

In order for the above lemma to be of any use, we need a connection between the h_is and the partial allocations that are produced in the first part of Algorithm 3 (lines 3-5). At a first glance, the issue is that the order in which the goods are assigned in Preprocessing(N, M) is somewhat different than the order in which the goods are allocated in Round-Robin$(N, \mathcal{A}, M, \ell, n)$. Next we establish this connection.

Lemma 4.2. *The partial allocation produced in line 3 of Algorithm 3 is $\mathcal{A} = (\{h_1\}, \{h_2\}, \ldots, \{h_n\})$, where the h_is are as in Lemma 4.1.*

Proof. We start with the easy observation that the ordering ℓ that is used in the Round-Robin subroutine in line 3 of Algorithm 3 *is not* the same with the order that goods get assigned to agents during the preprocessing step, even when one takes into account that moving agents into L is similar to changing their order.

First, using induction, we are going to show that agents in L get assigned to them (in Algorithm 4) the same goods they would if they were to choose first (in Algorithm 2) according to ℓ. For agent $\ell[1]$ this is straightforward, since she gets her favorite good in both cases. So, assume for our inductive step that agents $\ell[1], \ldots, \ell[k-1]$ did receive goods $h_{\ell[1]}, \ldots, h_{\ell[k-1]}$ respectively in Algorithm 2. We argue that agent $\ell[k]$'s favorite available good from $M_k = M \setminus \cup_{i=1}^{k-1} \{h_{\ell[i]}\}$ is $h_{\ell[k]}$. Indeed, in the execution of Algorithm 4, consider the last iteration of the *while* loop where $\ell[k]$ was the lexicographically first agent of A. At the time, L was $\{\ell[1], \ldots, \ell[k-1]\}$ and in lines 4-9 agent $\ell[k]$ gets assigned $h_{\ell[k]}$ which is her favorite good among the ones assigned outside L exactly because this was much better than her favorite unassigned good. That is, (the final choice for) $h_{\ell[k]}$ is $\ell[k]$'s favorite good from M_k. This concludes the inductive step.

Given the above, it is not hard to argue about agents in $N \setminus L$. First, observe that, on termination, agents in $N \setminus L$ all have distinct timestamps assigned in line 5. Although this is not true for all agents, an agent in $N \setminus L$ only gets an existing timestamp, if this belongs to one or more agents already in L. For the remainder of the proof, by *timestamp* of an agent i, we mean the *final value of t_i*. Now, fix some $i \in N \setminus L$. In Algorithm 4

agent i gets assigned her favorite good available if we exclude the goods assigned to *some*
agents in L, and to all the agents in $N \setminus L$ with timestamp less than t_i (*first scenario*). In
Algorithm 2, agent i receives her favorite good available if we exclude the goods allocated
to *all* the agents in L, and to all the agents in $N \setminus L$ with timestamp less than t_i (*second
scenario*). However, in the second scenario, the extra agents from L that pick before i
choose goods that i does not find attractive enough. Indeed, those goods are available in
the first scenario (where they are the only extra available options compared to the second
scenario), yet i does not prefer them to h_i. Therefore, in the second scenario i also picks
h_i and $A_i = \{h_i\}$ after line 3 of Algorithm 3. □

Given Lemmata 4.1 and 4.2, we are going to consistently use the h_i notation for
the goods allocated in line 3 of Algorithm 3, throughout the remaining of this section.
Further, for the agents who receive a second good in line 5 of Algorithm 3 we use h_i' to
denote that second good of agent i.

As a warm-up we first obtain that Algorithm 3 maintains the fairness guarantee of its
components, i.e., EF1 fairness.

Proposition 4.3. *Algorithm 3 returns an EF1 allocation.*

Proof. By Theorem 2.7, it suffices to show that the partial allocation $\mathcal{A} = (A_1, \ldots, A_n)$
produced in line 5 of Algorithm 3 is EF1. Fix two distinct agents $i, j \in N$. If $j \in L$, then
$A_j = \{h_j\}$. Clearly, $v_i(A_i) \geq v_i(A_j \setminus \{h_j\}) = 0$. On the other hand, if $j \in N \setminus L$, then
$A_j = \{h_j, h_j'\}$. Since agent i picked h_i when h_j' was still available, $v_i(h_i) \geq v_i(h_j')$. So,
we have $v_i(A_i) \geq v_i(h_i) \geq v_i(A_j \setminus \{h_j\})$. □

4.1. Envy-Freeness up to Any Good

Proving whether EFX allocations always exist or not seems very challenging [31, 13, 16].
Even improving on the $1/2$ approximation factor of Plaut and Roughgarden [31] has been
one of the most intriguing recent open problems in fair division. In this sense, we view
the following as one of the highlights of this work.

Theorem 4.4. *The allocation $\mathcal{A} = (A_1, \ldots, A_n)$ returned by Algorithm 3 is a $(\phi - 1)$-EFX
allocation.*

Proof. Consider the allocation $\mathcal{A} = (A_1, \ldots, A_n)$ returned by the algorithm, and fix two
distinct agents $i, j \in N$. If $|A_j| = 1$, then clearly, $v_i(A_i) \geq \max_{g \in A_j} v_i(A_j \setminus \{g\}) = 0$.
So, assume that $|A_j| \geq 2$ and let h be the last good added to A_j (either in line 5 by
reverse round-robin or in line 6 by envy-cycle-elimination). Of course, at the time this
happened, A_j may belonged to an agent j' other than j. Finally, let $A_i^{\text{old}}, A_{j'}^{\text{old}}$ be the
bundles of i and j', respectively, right before h was allocated (i.e., h was added to $A_{j'}^{\text{old}}$).
Note that A_i^{old} may not necessarily be a subset of A_i due to the possible swaps imposed
by Algorithm 1, but part b) of Theorem 2.7 implies that $v_i(A_i) \geq v_i(A_i^{\text{old}})$. We consider
four cases, depending on whether $i \in L$ and on the type of step during which h was added
to $A_{j'}^{\text{old}}$.

<u>Case 1 ($i \in L$ and h added in line 5).</u> We have $A_i^{\text{old}} = \{h_i\}$, as well as $j' \in N \setminus L$ and
$A_j = \{h_{j'}, h_{j'}'\}$. This immediately implies that $v_i(A_i^{\text{old}}) \geq \max\{v_i(h_{j'}), v_i(h_{j'}')\}$ and, thus,

11

354 $v_i(A_i) \geq \max_{g \in A_j} v_i(A_j \setminus \{g\})$.[2]

Case 2 ($i \in L$ and h added in line 6). By the way that envy-cycle-elimination chooses who to give the next good to, (line 10 of Algorithm 1), we know that right before h was added, no one envied j'. In particular, $v_i(A_i^{\text{old}}) \geq v_i(A_{j'}^{\text{old}})$. We further have $v_i(A_i^{\text{old}}) \geq v_i(h_i) > \phi \cdot v_i(h)$, where the last inequality directly follows from part a) of Lemma 4.1. Putting everything together,

$$v_i(A_j) = v_i(A_{j'}^{\text{old}}) + v_i(h) \leq (1 + \phi^{-1})v_i(A_i^{\text{old}}) \leq \phi \cdot v_i(A_i),$$

or, equivalently, $v_i(A_i) \geq \phi^{-1} \cdot v_i(A_j) = (\phi - 1)v_i(A_j)$.

Case 3 ($i \notin L$ and h added in line 5). We have $i, j' \in N \setminus L$ and $A_j = \{h_{j'}, h'_{j'}\}$. If $\ell[i] < \ell[j']$, then we proceed in a way similar to Case 1. Indeed,

$$v_i(A_i) \geq v_i(A_i^{\text{old}}) \geq v_i(h_i) \geq \max\{v_i(h_{j'}), v_i(h'_{j'})\} = \max_{g \in A_j} v_i(A_j \setminus \{g\}).$$

So, assume that $\ell[i] > \ell[j']$. This, in particular, means that $v_i(h'_i) \geq v_i(h'_{j'})$. We have

$$v_i(A_i) \geq v_i(A_i^{\text{old}}) \geq v_i(h_i) + v_i(h'_i) \geq \frac{1}{\phi} v_i(h_{j'}) + v_i(h'_{j'})$$

$$\geq \frac{1}{\phi}(v_i(h_{j'}) + v_i(h'_{j'})) = (\phi - 1)v_i(A_j),$$

where the third inequality directly follows from part b) of Lemma 4.1.

Case 4 ($i \notin L$ and h added in line 6). Arguing like in Case 2, we have $v_i(A_i^{\text{old}}) \geq v_i(A_{j'}^{\text{old}})$. Moreover, by the way round-robin works, we know that $v_i(h_i) \geq v_i(h'_i) \geq v_i(h)$. In particular, $v_i(h) \leq \frac{1}{2}v_i(\{h_i, h'_i\}) \leq \frac{1}{2}v_i(A_i^{\text{old}})$. Putting things together, we have

$$v_i(A_j) = v_i(A_{j'}^{\text{old}}) + v_i(h) \leq \left(1 + \frac{1}{2}\right)v_i(A_i^{\text{old}}) \leq \phi \cdot v_i(A_i).$$

357 Equivalently, $v_i(A_i) \geq (\phi - 1)v_i(A_j)$. $\qquad\square$

It is not hard to see that our analysis is tight, i.e., there are instances (even with $n = 2$ and $m = 4$) for which the resulting allocation is not $(\phi - 1 + \varepsilon)$-EFX for any $\varepsilon > 0$.

360 ### 4.2. Groupwise Maximin Share Fairness

A result of Amanatidis et al. [5] (Proposition 3.4) implies that every *exact* EFX allocation is also a 4/7-GMMS allocation. Of course, the allocation produced by Algorithm
363 3 is not exact EFX and, in general, an arbitrary $(\phi - 1)$-EFX allocation need not even be a 0.404-GMMS allocation (see Appendix B). For the particular allocation returned by Algorithm 3, however, we can show that the GMMS guarantee is significantly better.
366 Parts of our proof closely follow the proof of the aforementioned proposition of [5].

Before moving on to the technical details, it is worth mentioning that our analysis does not yield a stronger guarantee for the weaker notion of MMS. This is partly because

---

[2] Here we achieve the EFX objective exactly. Instead, in a similar argument as in Case 2, we could have used that $v_i(h_i) > v_i(h_{j'})$ and $v_i(h_i) > \phi \cdot v_i(h'_{j'})$ to get $v_i(A_i) \geq (\phi - 1)v_i(A_j)$.

it relies on guarantees for approximate EF or EFX (following from the proof of Theorem 4.4) which in turn hold for any subset of agents. Moreover, the algorithm itself is designed with EFX as the primary goal and, thus, it is lacking specific subroutines that are known to yield MMS guarantees, like bipartite matchings and bag-filling [26, 4, 7, 23, 24, 21].

We are going to need the following simple lemma that allows to remove appropriately chosen subsets of goods, while reducing the number of agents, so that the maximin share of a specific agent does not decrease. In particular, the lemma implies that for any good g, $\boldsymbol{\mu}_i(n-1, M \setminus \{g\}) \geq \boldsymbol{\mu}_i(n, M)$.

Lemma 4.5 (Amanatidis et al. [5]). *Suppose $\mathcal{T} \in \Pi_n(M)$ is an n-maximin share defining partition for agent i. Then, for any set of goods S, such that there exists some j with $S \subseteq T_j$, it holds that $\boldsymbol{\mu}_i(n-1, M \setminus S) \geq \boldsymbol{\mu}_i(n, M)$.*

Theorem 4.6. *The allocation $\mathcal{A} = (A_1, \ldots, A_n)$ returned by Algorithm 3 is a $\frac{2}{\phi+2}$-GMMS allocation.*

Proof. Suppose that $\mathcal{A}$ is not a $\frac{2}{\phi+2}$-GMMS allocation, i.e., there exists a subset of agents $Q \subseteq N$ with $|Q| = q$, and some agent $j \in Q$, so that $v_j(A_j) < \frac{2}{\phi+2}\boldsymbol{\mu}_j(q, R)$, where $R = \cup_{k \in Q} A_k$. That is, with respect to Q and R, the restriction of $\mathcal{A}$ to Q is not a $\frac{2}{\phi+2}$-MMS allocation. To facilitate the presentation, and without loss of generality, we may assume that $Q = [q]$ and that agent 1 is such a "dissatisfied" agent. We write $\boldsymbol{\mu}_1$ instead of $\boldsymbol{\mu}_1(q, R)$.

We may remove any agent in Q, *other* than agent 1, that receives exactly one good, and still end up with a suballocation that is not a $\frac{2}{\phi+2}$-GMMS allocation. Indeed, if $|A_i| = 1$ for some $i \in Q \setminus \{1\}$, then $(A_1, \ldots, A_{i-1}, A_{i+1}, \ldots, A_q)$ is an allocation of $R \setminus A_i$ to $Q \setminus \{i\}$ and, by Lemma 4.5, $\boldsymbol{\mu}'_1 = \boldsymbol{\mu}_1(q-1, R \setminus A_i) \geq \boldsymbol{\mu}_1$. Thus, $v_1(A_1) < \frac{2}{\phi+2}\boldsymbol{\mu}'_1$. Therefore, again without loss of generality, we may assume that $|A_i| \geq 2$ for all $i \in Q \setminus \{1\}$ in the initial allocation $\mathcal{A}$. At this point, we make the distinction on whether $1 \in L$ or not.

<u>Case 1 ($1 \in L$).</u> As we see from (the footnote of) Case 1 and from Case 2 of the proof of Theorem 4.4, we always have $v_1(A_1) \geq (\phi-1)v_1(A_i)$ (or equivalently $v_1(A_i) \leq \phi v_1(A_1)$) for all $i \in Q \setminus \{1\}$. Recall that, by the definition of maximin share, $\boldsymbol{\mu}_1 \leq \frac{1}{q}v_1(R)$. Thus

$$q\boldsymbol{\mu}_1 \leq v_1(R) = \sum_{k \in Q} v_1(A_k) \leq q\phi v_1(A_1).$$

That is, we get $v_1(A_1) \geq (\phi-1)\boldsymbol{\mu}_1 \geq \frac{2}{\phi+2}\boldsymbol{\mu}_1$, which contradicts the choices of $\mathcal{A}$ and A_1.

<u>Case 2 ($1 \notin L$).</u> Consider some $i \in Q \setminus \{1\}$ and let h be the last good added to A_i. Following the notation introduced in the proof of Theorem 4.4, this bundle belonged to some agent i' and $A_{i'}^{\text{old}}$ denotes the bundle allocated to i' right before h was added. According to Cases 3 and 4 in the proof of Theorem 4.4, if h was added in line 5 of Algorithm 3 and $\ell[1] \geq \ell[i']$ or if it was added in line 6, then $v_1(A_i) \leq \phi v_1(A_1)$. We still need to deal with the subcase where h was added in line 5 but $\ell[1] < \ell[i']$. We call such an A_i *dubious*. For dubious bundles, by their definition, we directly have $|A_i| = 2$ and $v_1(A_1) \geq v_1(h_1) \geq \max_{g \in A_i} v_1(g)$. If a bundle A_i is not dubious, or if it is dubious but we have $v_1(A_i) \leq \frac{3}{2}v_1(A_1) < \phi v_1(A_1)$, we say that A_i is *convenient*. A (dubious)

bundle is *inconvenient* if it is not convenient. A good is inconvenient if it belongs to an inconvenient bundle. Let B be the set of all inconvenient goods.

Now we are going to show that $v_1(A_1) \geq \frac{2}{\phi+2}\boldsymbol{\mu}_1(q', R')$ for a reduced instance that we get by possibly removing some inconvenient goods. We do so in a way that ensures that $\boldsymbol{\mu}_1(q', R') \geq \boldsymbol{\mu}_1$, thus contradicting the choices of $\mathcal{A}$ and A_1. We consider a q-maximin share defining partition $\mathcal{T}$ for agent 1 with respect to R, i.e., $\min_{T_i \in \mathcal{T}} v_1(T_i) = \boldsymbol{\mu}_1$ and $\cup_{k \in Q} T_k = R$.[3] If there is a bundle of $\mathcal{T}$ containing two goods of B, g_1, g_2, then we remove those two goods and reduce the number of agents by one. By Lemma 4.5, we have that $\boldsymbol{\mu}_1(q-1, R \setminus \{g_1, g_2\}) \geq \boldsymbol{\mu}_1$. We repeat as many times as necessary to get a reduced instance with $q' \leq q$ agents and a set of goods $R' \subseteq R$ for which there is a q'-maximin share defining partition $\mathcal{T}'$ for agent 1, such that no bundle contains more than *one* good from B. By repeatedly using Lemma 4.5, we get $\boldsymbol{\mu}_1(q', R') \geq \boldsymbol{\mu}_1$.

Let x be the number of goods from B in the reduced instance. Clearly, x cannot be greater than q', or some bundle of $\mathcal{T}'$ would contain at least 2 inconvenient goods. Further, if $|B| = y$, i.e., the number of inconvenient goods in the original instance, then we know that the number of convenient bundles in the restriction of $\mathcal{A}$ on Q was $q - \frac{y}{2}$, and that the number of agents was reduced $\frac{y-x}{2}$ times, i.e., $q' = q - \frac{y-x}{2}$. That is, we can express the number of convenient bundles in the original instance in terms of q' and x only, as $q' - \frac{x}{2}$.

In order to upper bound $v_1(R')$, notice that R' contains all the goods of all the convenient bundles plus x inconvenient goods. Recall that any good of a dubious bundle has value at most $v_1(A_1)$, and that if A_i is convenient then $v_1(A_i) \leq \phi v_1(A_1)$. So, we have

$$\begin{aligned}
v_1(R') &\leq x v_1(A_1) + \left(q' - \frac{x}{2} - 1\right)\phi v_1(A_1) + v_1(A_1) \\
&= (\phi q' + (1 - \phi/2)x - (\phi-1))\, v_1(A_1) \\
&\leq (\phi + (1 - \phi/2))\, q' v_1(A_1) = \frac{\phi+2}{2} q' v_1(A_1).
\end{aligned}$$

Combining this inequality with $\boldsymbol{\mu}_1 \leq \boldsymbol{\mu}_1(q', R')$ (by the construction of the reduced instance) and $\boldsymbol{\mu}_1(q', R') \leq \frac{1}{q'} v_1(R')$ (by the definition of maximin share), we get $v_1(A_1) \geq \frac{2}{\phi+2}\boldsymbol{\mu}_1$, which contradicts the choices of $\mathcal{A}$ and A_1. $\qquad \square$

A natural question is why we do not achieve the factor of $4/7$ of the original proposition of [5] instead. A close inspection of the original proof reveals that we need a slightly stronger upper bound for the value of the convenient bundles, i.e., a factor of $3/2$ rather than ϕ that we actually have here. There is no easy way to fix this *in general* without other things breaking down badly in the analysis of Algorithm 3. The crucial observation, however, is that we only need the distinction of convenient and inconvenient bundles for agents in $N \setminus L$. By fine-tuning line 8 of Algorithm 4, we are able to improve the inequalities about the convenient bundles just for agents in $N \setminus L$ and obtain a $4/7$-GMMS allocation, at the expense of some loss with respect to EFX.

---

[3] Note that while goods in B may have no apparent significance for the reduced instances and the allocations we talk about from this point on, we keep referring to them as *inconvenient*.

Theorem 4.7. *Suppose we modified Algorithm 3 by changing ϕ in line 8 of Algorithm 4 to $3/2$. Then the resulting allocation is a $4/7$-GMMS allocation. It is also a $3/5$-EFX, a $2/3$-PMMS, and an EF1 allocation.*

Proof. The proof for EF1 goes through as is. We only need to highlight the differences in comparison to the proofs of Theorems 4.4, 4.9 and 4.6. We begin with EFX and Theorem 4.4. When $|A_j| = 1$ it is again straightforward that $v_i(A_i) \geq \max_{g \in A_j} v_i(A_j \setminus \{g\})$. In Cases 1 and 2 (using the footnote of Case 1), by substituting $2/3$ for ϕ, we get $v_i(A_i) \geq \frac{3}{5} v_i(A_j)$. The first part of Case 3 (where $\ell[i] < \ell[j']$) is the same, giving $v_i(A_i) \geq \max_{g \in A_j} v_i(A_j \setminus \{g\})$, while in the second part (where $\ell[i] > \ell[j']$) we get $v_i(A_i) \geq \frac{2}{3} v_i(A_j)$ by using $3/2$ instead of ϕ. Case 4 is exactly the same, giving $v_i(A_i) \geq \frac{2}{3} v_i(A_j)$. So, for any pair i, j of agents, we either have $v_i(A_i) \geq \max_{g \in A_j} v_i(A_j \setminus \{g\})$ or $v_i(A_i) \geq \frac{3}{5} v_i(A_j)$, implying that $v_i(A_i) \geq \frac{3}{5} \max_{g \in A_j} v_i(A_j \setminus \{g\})$.

Moving to PMMS and Theorem 4.9, by following the same proof and using the $3/5$ factor from above instead of ϕ in the last couple of lines, we still get the same factor $2/3$.

Finally, for GMMS and Theorem 4.6, we may follow the same arguments with $4/7$ rather than $\phi/(\phi + 2)$. For Case 1, using the inequality $v_1(A_1) \geq \frac{3}{5} v_1(A_i)$ from the EFX case above, we have $v_1(A_1) \geq \frac{3}{5} \boldsymbol{\mu}_1 \geq \frac{4}{7} \boldsymbol{\mu}_1$. For Case 2, the above analysis for EFX implies that when $1 \notin L$ and A_i is not dubious we have the even stronger guarantee: $v_1(A_1) \geq \frac{2}{3} v_1(A_i)$. This directly implies that $v_1(A_1) \geq \frac{2}{3} v_1(A_i)$ for any *convenient* bundle A_i. Coming to the final argument for upper bounding $v_1(R')$, we have

$$v_1(R') \leq x v_1(A_1) + \left(q' - \frac{x}{2} - 1 \right) \frac{3}{2} v_1(A_1) + v_1(A_1)$$
$$= \left(\frac{3q'}{2} + \frac{x}{4} - \frac{1}{2} \right) v_1(A_1) \leq \frac{7}{4} q' v_1(A_1).$$

Like before, we combine this inequality with $\boldsymbol{\mu}_1 \leq \boldsymbol{\mu}_1(q', R')$ and $\boldsymbol{\mu}_1(q', R') \leq \frac{1}{q'} v_1(R')$ to get that $v_1(A_1) \geq \frac{4}{7} \boldsymbol{\mu}_1$. □

Remark 4.8. As we already mentioned, our approach in both theorems is heavily based on the EF or EFX guarantees we get in the various different cases and depends less on the details of the algorithms themselves. For that reason the analyses are probably not tight. On the other hand, for any $\varepsilon > 0$, there are instances for which Algorithm 3 does not return a $(2/3 + \varepsilon)$-GMMS allocation, independently of how we may tune the constant in line 8 of Algorithm 4. While we omit the description of such instances since they do not establish tightness of the algorithm, we suspect that the actual approximation ratio of the modified Algorithm 3 in Theorem 4.7 for GMMS is $2/3$.

4.3. Pairwise Maximin Share Fairness

Any result for GMMS directly translates to a result for PMMS with the exact same guarantee. Note, however, that the proof of Theorem 4.6 suggests that the bad event with respect to GMMS is having many inconvenient bundles. When we only deal with two agents at a time, it is not hard to see that inconvenient bundles are not an issue. In fact, their existence would not be able to force the approximation ratio for PMMS below $2/3$, even if the goods where divisible. Indeed, following the cases in the proof of Theorem 4.4 it is relatively easy to show that this is exactly the guarantee achieved by Algorithm 3.

Theorem 4.9. *The allocation $\mathcal{A} = (A_1, \ldots, A_n)$ returned by Algorithm 3 is a $2/3$-PMMS allocation.*

Proof. Fix two distinct agents $i, j \in N$. To show the desired approximation ratio we are going to use the analysis in the proof of Theorem 4.4 and the simple inequality $2\boldsymbol{\mu}_i(2, S) \leq v_i(S)$ that directly follows from Definition 2.4 for any subset S of goods. To simplify the notation, we use A_{ij} as a shorthand for $A_i \cup A_j$.

If $|A_j| = 1$, then by Lemma 4.5, $\boldsymbol{\mu}_i(2, A_{ij}) \leq \boldsymbol{\mu}_i(1, A_i) = v_i(A_i)$. So, assume that $|A_j| \geq 2$ and let h be the last good added to A_j. We adopt the notation of the proof of Theorem 4.4 here as well. That is, we assume that right before h was added, A_j belonged to some agent j'. We first examine the case where $i \in N \setminus L$, h was added in line 5 of Algorithm 3 by reverse round-robin, and $\ell[i] < \ell[j']$. This case is the *bottleneck* for achieving a better approximation factor, as the rest of the analysis reveals. Here we know that $A_j = \{h_{j'}, h'_{j'}\}$ and that $\max\{v_i(h_{j'}), v_i(h'_{j'})\} \leq v_i(h_i)$. Therefore,

$$2\boldsymbol{\mu}_i(2, A_{ij}) \leq v_i(A_{ij}) \leq v_i(A_i) + 2v_i(h_i) \leq 3v_i(A_i).$$

It immediately follows that $v_i(A_i) \geq \frac{2}{3}\boldsymbol{\mu}_i(2, A_{ij})$.

For all the other cases, by examining the proof of Theorem 4.4, we can see that $v_i(A_j) \leq \phi v_i(A_i)$. Therefore,

$$2\boldsymbol{\mu}_i(2, A_{ij}) \leq v_i(A_{ij}) \leq v_i(A_i) + \phi v_i(A_i) \leq 3v_i(A_i).$$

Again, it follows that $v_i(A_i) \geq \frac{2}{3}\boldsymbol{\mu}_i(2, A_{ij})$. $\qquad\square$

While the above factor is tight for our algorithm, it is easy to see that if we exclude the bottleneck case in the proof of Theorem 4.9, then the approximation ratio goes up to $\frac{2}{\phi+1} \approx 0.764$. Hence, we could try to improve this single problematic case where both i and j' receive two goods from the round-robin subroutine but j' has lower priority. Note that the bundles of i and j' start off well, i.e., right after line 5 of Algorithm 3 we have $v_i(A_i) \geq \boldsymbol{\mu}_i(2, A_{ij'})$. The issue is that during the envy-cycle-elimination phase, A_i might be updated to a bundle that still has value almost $v_i(h_i)$ but can be combined with $h_{j'}$ and $h'_{j'}$ to produce two sets of value roughly $\frac{3}{2}v_i(h_i)$. To remedy that, we can modify slightly the envy graph. The high level idea—due to Kurokawa [25]—is that an agent from $N \setminus L$ should only exchange her *initial* bundle of two goods for something significantly better.

Suppose we start with a partial allocation $\mathcal{P} = (P_1, \ldots, P_n)$ produced in line 5 of Algorithm 3. For $\alpha > 1$, the α-modified envy graph $G_{\mathcal{P}}^\alpha$ is defined like the envy graph $G_{\mathcal{P}}$ but we drop any edge (i, j) where: $i \in N \setminus L$, *and* i still has her original bundle, *and* $\alpha \cdot v_i(P_i) > v_i(P_j)$. That is, agents in $N \setminus L$ are represented in the envy graph as having an artificially amplified value (by a factor of α) specifically for their original bundles.

The following theorem indicates how far we can push the approximation factor for PMMS, at the expense of EF1, while preserving the original guarantees with respect to EFX and GMMS.

Theorem 4.10. *Suppose we modified Algorithm 3 by using the $(\phi - \frac{1}{2})$-adjusted envy graph in Algorithm 1. Then the resulting allocation is a $\frac{4\phi-2}{2\phi+3}$-PMMS and a $\frac{2}{2\phi-1}$-EF1 allocation. Moreover, the guarantees of Theorems 4.4 and 4.6 are not affected.*

16

Proof. First, it is easy to see the EF1 guarantee. Before running the modified envy-cycle-elimination algorithm, the allocation is EF1. Then, whenever a new good gets allocated, it is given to an agent that no one envies by more than a factor of $\frac{2\phi-1}{2}$. That is, for any agents $i, j \in N$, if h is the last good given to j, then $v_i(A_i) \geq \frac{2}{2\phi-1} v_i(A_j \setminus \{h\})$.

For the other notions, we only need to highlight the differences from the proofs of Theorems 4.4, 4.9 and 4.6. We begin with EFX and Theorem 4.4. It is easy to see that the *only* step that differs is Case 4. It is not anymore the case that $v_i(A_i^{\mathrm{old}}) \geq v_i(A_{j'}^{\mathrm{old}})$. Instead, $(\phi - \frac{1}{2})v_i(A_i^{\mathrm{old}}) \geq v_i(A_{j'}^{\mathrm{old}})$. Also, like in the original proof, we have $v_i(h) \leq \frac{1}{2}v_i(A_i^{\mathrm{old}})$. Putting everything together, we have

$$v_i(A_j) = v_i(A_{j'}^{\mathrm{old}}) + v_i(h) \leq \left(\phi - \frac{1}{2} + \frac{1}{2}\right)v_i(A_i^{\mathrm{old}}) = \phi \cdot v_i(A_i).$$

We conclude that we have a $(\phi - 1)$-EFX allocation.

Given the guarantee for EFX, the proof of Theorem 4.6 for GMMS goes through as is.

Finally, for PMMS and Theorem 4.9 we may follow the same general proof except for some details. We fix two agents $i, j \in N$. Like in the proof of Theorem 4.9 we use A_{ij} to denote $A_i \cup A_j$. If $|A_j| = 1$, the original argument holds, so we assume that $|A_j| \geq 2$. Let h be the last good added to A_j and assume that right before h was added, A_j belonged to some agent j'.

Again, we first go over the bottleneck case where $i \in N \setminus L$, h was added in line 5 of Algorithm 3 by reverse round-robin, and $\ell[i] < \ell[j']$. Then $A_j = \{h_{j'}, h'_{j'}\}$ and $\max\{v_i(h_{j'}), v_i(h'_{j'})\} \leq v_i(h_i)$. A vital distinction now is whether A_i is i's original bundle from line 5 of Algorithm 3. Suppose this is the case, i.e., $A_i = \{h_i, h'_i\}$ and $A_{ij} = \{h_i, h'_i, h_j, h'_j\}$. Then, $h_i = \arg\max_{g \in A_{ij}} v_i(g)$ and it is not hard to see that $v_i(A_i) \geq \boldsymbol{\mu}_i(2, A_{ij})$ (see also Lemma 5.3 i) within the proof of Theorem 5.1). Next, suppose A_i is not i's original bundle. Then, for the modified envy-cycle-elimination algorithm to give i another bundle, it must be the case that $v_i(A_i) \geq \frac{2\phi-1}{2} v_i(\{h_i, h'_i\})$. Therefore,

$$2\boldsymbol{\mu}_i(2, A_{ij}) \leq v_i(A_{ij}) \leq v_i(A_i) + 2v_i(h_i) \leq v_i(A_i) + 2v_i(\{h_i, h'_i\})$$
$$\leq v_i(A_i) + \frac{4}{2\phi-1} v_i(A_i) = \frac{2\phi+3}{2\phi-1} v_i(A_i).$$

It follows that $v_i(A_i) \geq \frac{4\phi-2}{2\phi+3} \boldsymbol{\mu}_i(2, A_{ij})$.

For all the other cases, we have that $v_i(A_j) \leq \phi v_i(A_i)$. Therefore, $2\boldsymbol{\mu}_i(2, A_{ij}) \leq v_i(A_{ij}) \leq v_i(A_i) + \phi v_i(A_i)$ and it follows that $v_i(A_i) \geq \frac{2}{\phi+1} \boldsymbol{\mu}_i(2, A_{ij}) \geq \frac{4\phi-2}{2\phi+3} \boldsymbol{\mu}_i(2, A_{ij})$.

We conclude that the allocation is $\frac{4\phi-2}{2\phi+3}$-PMMS. □

5. GMMS, PMMS, and EFX with a Few Goods

In this section we focus on the exact versions of the fairness notions under consideration. In particular, we show that GMMS allocations always exist when $m \leq n + 2$. This implies that PMMS and EFX allocations also exist for this case by the discussion in Section 2.[4]

---

[4] Actually, the existence of EFX allocations is directly implied by the existence of PMMS allocations *only* when all values are positive. However, our result is more general.

As it is indicated in the proof of Theorem 5.1, the interesting case is when $m = n + 2$ and is tackled by Algorithm 5. When $m \leq n$ the problem is trivial, and the $m = n + 1$ case is rather straightforward as well. Adding one extra good, however, makes things significantly more complex. To point out how challenging these simple restricted cases can be, we note that for the much better studied notion of MMS fairness it is still open whether exact MMS allocations exist when $m = n + 5$ [26].

Quite surprisingly, the envy-cycle-elimination algorithm again comes to rescue for the case when $m = n + 2$. We first run the round-robin algorithm to allocate $n - 1$ goods to the first $n - 1$ agents. After this, we have 3 goods remaining. Allocating these goods to the last agent may destroy the properties we are after, so we need to be careful on how to handle these three goods. Instead, we (pretend to) pack them into two boxes; the big box (i.e., the virtual good p) "contains" two goods and the small box (i.e., the virtual good q) "contains" one. We tell each agent separately that the big box contains her favorite two out of the three items and give the big box to the last agent. Then we proceed using the envy-cycle-elimination algorithm. At the end, the owner of the big box gets her two favorite goods, while the owner of the small box gets the remaining good.

Algorithm 5: Draft-Pack-and-Eliminate(N, M)

---
1 Let $\ell = (1, 2, \ldots, n)$ and $\mathcal{A} = (\emptyset, \ldots, \emptyset)$
2 $(\mathcal{A}, M') = $ Round-Robin$(N, \mathcal{A}, M, \ell, n - 1)$
3 Create two virtual goods p and q, such that for all $i \in N$:
   $v_i(q) = \min_{g \in M'} v_i(g)$ and $v_i(p) = v_i(M') - v_i(q)$
4 Allocate p to agent n
5 $\mathcal{A} = $ Envy-Cycle-Elimination$(N, \mathcal{A}, \{q\})$
6 Give to the final owner of p her two favorite goods from M' and to the final owner of q the remaining good
7 **return** $\mathcal{A}$

Theorem 5.1. *For instances with $m \leq n + 2$, a GMMS allocation always exists and can be efficiently computed.*

Proof. When $m \leq n$, we arbitrarily allocate one good to each agent, till there are no goods left, to produce $\mathcal{A} = (A_1, \ldots, A_n)$. Fix a subset S of agents, and let $i \in N$. The combined bundle of all agents in S either contains strictly less than $|S|$ goods or exactly $|S|$ goods. In the first case, we trivially have $\boldsymbol{\mu}_i(|S|, \cup_{j \in S} A_j) = 0$, whereas in the second case, we have $\boldsymbol{\mu}_i(|S|, \cup_{j \in S} A_j) = \min_{g \in \cup_{j \in S} A_j} v_i(g)$. In both cases, we have that $v_i(A_i) \geq \boldsymbol{\mu}_i(|S|, \cup_{j \in S} A_j)$, and $\mathcal{A}$ is a GMMS allocation.

For $m > n$, we will use the following simple observations.

Lemma 5.2. *Let $Q \subseteq N$ and $T \subseteq M$ such that $|T| = 2|Q| - 1$. Then, for any $i \in Q$, we have $\max_{g \in T} v_i(g) \geq \boldsymbol{\mu}_i(|Q|, T)$.*

Proof of Lemma 5.2. By the pigeonhole principle, in any possible partition of T in $|Q|$ parts, at least one bundle will have at most 1 good. So, in any $|Q|$-maximin share partition of T for an agent $i \in Q$, her worst bundle's worth is upper bounded by $\max_{g \in T} v_i(g)$. $\square$

The next one is established within the proof of Theorem 5.1 of [2].

18

Lemma 5.3. *Let $i \in N$ and $T = \{g_1, g_2, g_3, g_4\} \subseteq M$ such that $v_i(g_1) \geq v_i(g_2) \geq v_i(g_3) \geq v_i(g_4)$. Then*

  i) $v_i(\{g_1, g_4\}) \geq \boldsymbol{\mu}_i(2, T)$ *, and*

  ii) $\max\{v_i(g_1), v_i(\{g_2, g_3\})\} \geq \boldsymbol{\mu}_i(2, T)$.

When $m = n+1$, let $\mathcal{A} = (\emptyset, \ldots, \emptyset)$, i.e., $A_i = \emptyset$ for all $i \in N$, and $\ell = (1, 2, \ldots, n)$, i.e., the standard lexicographic order. We can first run Round-Robin$(N, \mathcal{A}, M, \ell, n)$ and then give the remaining good to agent n to get the final allocation $(\{h_1\}, \ldots, \{h_{n-1}\}, \{h_n, h'_n\})$. Consider a subset of agents S and an agent $i \in S$. From S, we can eliminate anyone, other than i, that owns at most 1 good, without reducing i's maximin share. That is, $v_i(A_i) = \boldsymbol{\mu}_i(1, A_i) \geq \boldsymbol{\mu}_i(|S|, \cup_{j \in S} A_j)$, where the last inequality follows after $|S| - 1$ applications of Lemma 4.5. Hence, if no agent in $S \setminus \{i\}$ got 2 goods, then we would be done. The problem then reduces to checking the case where S consists of agents i and n. In particular, we need to show that $v_i(h_i) \geq \boldsymbol{\mu}_i(2, \{h_i, h_n, h'_n\})$ for all $i < n$. Indeed, given that $v_i(h_i) \geq \max\{v_i(h_n), v_i(h'_n)\}$, this directly follows from Lemma 5.2. Hence, $\mathcal{A}$ is a GMMS allocation.

When $m = n + 2$, we use Algorithm 5 to compute the allocation $\mathcal{A} = (A_1, \ldots, A_n)$. We consider two cases, depending on whether $\mathcal{A}$ contains a bundle with 3 goods or not.

<u>Case 1 (One agent receives 3 goods).</u> Let $\hat{\jmath}$ be the agent for whom $|A_{\hat{\jmath}}| = 3$. By arguing like before about repeatedly eliminating agents that received exactly 1 good via Lemma 4.5, it is easy to see that the problem of whether $v_i(A_i) \geq \boldsymbol{\mu}(|S|, \cup_{j \in S} A_j)$, for any $S \subseteq N$ and any $i \in S$, is reduced to whether $v_i(A_i) \geq \boldsymbol{\mu}_i(2, A_i \cup A_{\hat{\jmath}})$ for any $i \in N \setminus \{\hat{\jmath}\}$.

The only way that $\hat{\jmath}$ ended up with 3 goods is if she received both p and q and $A_{\hat{\jmath}} = M'$. When q was allocated, some agent $\hat{\jmath}'$ (possibly other than $\hat{\jmath}$) had p. Given that no one envied $\hat{\jmath}'$ at the time and that the envy-cycle-elimination never decreases the value of an agent's bundle (Theorem 2.7 b)), for any $i \in N \setminus \{\hat{\jmath}\}$ we have

$$v_i(A_i) \geq v_i(p) = v_i(A_{\hat{\jmath}}) - \min_{g \in A_{\hat{\jmath}}} v_i(g) = \max_{g \in A_{\hat{\jmath}}} v_i(A_{\hat{\jmath}} \setminus \{g\}).$$

Using part ii) of Lemma 5.3, this implies that $v_i(A_i) \geq \boldsymbol{\mu}_i(2, A_i \cup M')$. So, in this case $\mathcal{A}$ is a GMMS allocation.

<u>Case 2 (Two agents receive 2 goods each).</u> Let $\hat{\jmath}_p$ be the agent who ended up with p and $\hat{\jmath}_q$ be the agent who ended up with q (in addition to her other good d) after line 5 of Algorithm 5. Clearly, these are the only agents who receive 2 goods. Further, let $A_{\hat{\jmath}_p} = \{a, b\}$ and $A_{\hat{\jmath}_2} = \{c, d\}$, i.e., $M' = \{a, b, c\}$ and a, b are $\hat{\jmath}_p$'s two favourite goods from M'.

Again, by repeatedly using Lemma 4.5, the problem reduces to a small number of subcases involving at most one agent that received 1 good. Specifically, to show that $\mathcal{A}$ is a GMMS allocation, it suffices to show

  (i) $v_i(A_i) \geq \boldsymbol{\mu}_i(2, A_i \cup A_{\hat{\jmath}_p})$ for any $i \in N \setminus \{\hat{\jmath}_p, \hat{\jmath}_q\}$

  (ii) $v_i(A_i) \geq \boldsymbol{\mu}_i(2, A_i \cup A_{\hat{\jmath}_q})$ for any $i \in N \setminus \{\hat{\jmath}_p, \hat{\jmath}_q\}$

  (iii) $v_i(A_i) \geq \boldsymbol{\mu}_i(3, A_i \cup A_{\hat{\jmath}_q} \cup A_{\hat{\jmath}_q})$ for any $i \in N \setminus \{\hat{\jmath}_p, \hat{\jmath}_q\}$

  (iv) $v_{\hat{\jmath}_q}(A_{\hat{\jmath}_q}) \geq \boldsymbol{\mu}_{\hat{\jmath}_q}(2, A_{\hat{\jmath}_p} \cup A_{\hat{\jmath}_q})$

  (v) $v_{\hat{\jmath}_p}(A_{\hat{\jmath}_p}) \geq \boldsymbol{\mu}_{\hat{\jmath}_p}(2, A_{\hat{\jmath}_p} \cup A_{\hat{\jmath}_q})$

19

We first deal with subcases (i), (ii) and (iii). By how round-robin works, we know that i preferred her initial good to any good in M'. Moreover, by how envy-cycle-elimination works, we know that i did not envy the bundle $\{d\}$ right before q was added to it and that her current good is no worse than any good she had earlier. Thus, i's current good is her favourite good in $A_i \cup M'$. Then (i), (ii) and (iii) all follow by Lemma 5.2.

We next move to subcase (iv). Like above, because of round-robin, we know that $\hat{\jmath}_q$ preferred her initial good to any good in M' and, because of envy-cycle-elimination, her good d is no worse than her initial good. Thus, d is $\hat{\jmath}_q$'s favourite good in $A_{\hat{\jmath}_p} \cup A_{\hat{\jmath}_q}$. Even if c is her least favourite good, (iv) directly follows from part i) of Lemma 5.3.

Finally, we consider subcase (v). If $A_{\hat{\jmath}_p}$ contains $\hat{\jmath}_p$'s favourite good in $A_{\hat{\jmath}_p} \cup A_{\hat{\jmath}_q}$, then (v) follows from part i) of Lemma 5.3. So, suppose that $\hat{\jmath}_p$'s favourite good is either c or d. By line 6 of Algorithm 5, we know that $\min\{v_{\hat{\jmath}_p}(a), v_{\hat{\jmath}_p}(b)\} \geq v_{\hat{\jmath}_p}(c)$. Thus, it must be that d is $\hat{\jmath}_p$'s favourite good and that a, b are her second and third favourite goods. Moreover, $\hat{\jmath}_p$ did not envy the bundle $\{d\}$ right before q was added to it and $\{a, b\}$ is no worse than any bundle she had earlier. That is, $v_{\hat{\jmath}_p}(\{a, b\}) \geq v_{\hat{\jmath}_p}(d)$. Then (v) follows from part ii) of Lemma 5.3.

So, $\mathcal{A}$ is a GMMS allocation in this case as well. $\qquad\square$

Corollary 5.4. *When $m \leq n + 2$, we can efficiently find PMMS and EFX allocations.*

Proof. For PMMS this is trivial. Given that Caragiannis et al. [14] show that each PMMS allocation is an EFX allocation when all values are positive, we directly get the existence of EFX allocations for this special case. However, going through the proof of Theorem 5.1, it is not hard to see that the result holds under the stronger Definition 2.2 b).

Fix two distinct agents $i, j \in N$. As usual, if $|A_j| = 1$, then $v_i(A_i) \geq \max_{g \in A_j} v_i(A_j \setminus \{g\}) = 0$, so assume that $|A_j| \geq 2$.

First consider the case where $|A_j| = 2$ and let $|A_j| = \{x, y\}$. If j is the agent who ended up with p, then we directly have $v_i(A_i) \geq v_i(h_i) \geq \max\{v_i(x), v_i(y)\}$. So suppose j is the agent who ended up with $q = x$ in addition to her other good y. First, notice that $v_i(A_i) \geq v_i(h_i) \geq v_i(x)$. Because Algorithm 1 added x to $\{y\}$, we also get that $v_i(A_i) \geq v_i(y)$. We conclude that the allocation is EFX.

We then consider the case where $|A_j| = 3$. That is, j received both p and q and $A_j = M'$. But since Algorithm 1 added q to p, we have

$$v_i(A_i) \geq v_i(p) = v_i(M') - v_i(q) = v_i(A_j) - \min_{g \in A_j} v_i(g).$$

Again, the allocation is EFX. $\qquad\square$

Acknowledgments

Appendix A. An Example Illustrating the Different Notions

Suppose we have the following instance with three agents and five goods. It can be verified that this instance does not admit an EF allocation.

	a	b	c	d	e
Agent 1	10	6	7	5	3
Agent 2	6	8	12	7	5
Agent 3	10	11	3	2	7

First consider the allocation $\mathcal{A} = (A_1, A_2, A_3) = (\{a,d\}, \{c\}, \{b,e\})$. This is not an envy-free allocation, since $v_2(A_2) < v_2(A_1)$. However, we claim that it satisfies all the relaxed fairness criteria.

- To see that $\mathcal{A}$ is EFX for agent 1, we have $v_1(A_1) = 15$ and, on the other hand, $v_1(A_2 \setminus \{c\}) = 0$, $v_1(A_3 \setminus \{b\}) = 3$ and $v_1(A_3 \setminus \{e\}) = 6$. Thus, the EFX inequalities for agent 1 are satisfied. Similarly, we can check that for agents 2 and 3 the corresponding inequalities are satisfied as well.

- To verify that $\mathcal{A}$ is MMS, PMMS, and GMMS, we first observe that $\boldsymbol{\mu}_1(3, M) = 10$, $\boldsymbol{\mu}_2(3, M) = 12$ and $\boldsymbol{\mu}_3(3, M) = 10$. Hence, we have that $v_i(A_i) \geq \boldsymbol{\mu}_i(3, M)$ for $i = 1, 2, 3$, which establishes MMS. We can also examine the three different pairs of agents and see that the pairwise maximin shares are attained. For instance, $v_1(A_1) \geq \boldsymbol{\mu}_1(A_1 \cup A_2) = 10$, and $v_1(A_1) \geq \boldsymbol{\mu}_1(A_1 \cup A_3) = 11$. Hence, $\mathcal{A}$ is a PMMS allocation. As there is no other subset of agents to examine, this means $\mathcal{A}$ is also GMMS.

To demonstrate the approximate versions of the fairness criteria, consider now the allocation $\mathcal{A}' = (A_1', A_2', A_3') = (\{b\}, \{c,e\}, \{a,d\})$.

- $\mathcal{A}'$ is neither EF nor EFX, but it is easy to check that it is EF1. We also claim that it is a 0.6-EFX allocation. To see this, note that the approximation is due to agent 1, since agents 2 and 3 do not experience any envy. Observe that $v_1(A_1') = 0.6 \cdot v_1(A_3' \setminus \{d\}) \geq v_1(A_3' \setminus \{a\})$, and that $v_1(A_1') \geq 0.6 \cdot v_1(A_2' \setminus \{e\}) \geq 0.6 \cdot v_1(A_2' \setminus \{c\})$. Hence, indeed, $\mathcal{A}'$ is a 0.6-EFX allocation.

- It is also easy to verify that $\mathcal{A}'$ is a 0.6-MMS allocation, and the same approximation holds for PMMS and GMMS. Again it suffices to see the performance of agent 1, and we have $v_i(A_1') \geq 0.6\boldsymbol{\mu}_1(3, M)$, implying the approximation for MMS. It also holds that $v_1(A_1') = 0.6 \cdot \boldsymbol{\mu}_1(A_1' \cup A_3') \geq 0.6 \cdot \boldsymbol{\mu}_1(A_1' \cup A_2')$, which determines the approximation for PMMS and GMMS.

Appendix B. On the Connections Between the Approximate Versions of Fairness Guarantees

Here we deal with the question of whether the approximation guarantee established for one of the considered fairness notions directly implies the approximation guarantees for the other notions. For example, given that we obtain a $(\phi - 1)$-EFX approximation allocation, does this immediately enforce any of the fairness guarantees that we show for EF1, PMMS and GMMS? If this were true, our task would become simpler. Some of

our proofs would not be necessary and we would not have to analyze separately the approximation guarantees for all the fairness notions.

Such questions have been studied recently by Amanatidis et al. [5] who gave an almost complete picture regarding the relations among the notions of EF1, EFX, MMS, and PMMS.

We exhibit here that none of our improved approximation factors can be derived by a black-box use of the guarantees provided in [5]. Hence, the separate analyses of Algorithm 3 for each notion are necessary for obtaining our results.

To begin with, suppose we have an EF1 allocation. The results in Section 3 of Amanatidis et al. [5] state that an arbitrary EF1 allocation cannot yield a constant factor approximation for EFX and MMS—hence neither for GMMS. It is also shown that every EF1 allocation is 1/2-PMMS, and this is tight, but this guarantee is weaker than what we obtain here for PMMS.

Moving on to approximate EFX allocations, Proposition 3.7 in [5] states that an arbitrary α-EFX allocation is also a $2\alpha/(2+\alpha)$-PMMS allocation, and this is tight. This means that our $(\phi - 1)$-EFX approximation in Theorem 4.4 immediately implies only a $2(\phi - 1)/(1 + \phi) \approx 0.472$-PMMS approximation, which is worse than the guarantee in Theorem 4.9. Further, Proposition 3.5 in [5] implies (via MMS) that an arbitrary α-EFX allocation is not necessarily a $\max\{\frac{\alpha}{1+\alpha}, \frac{8\alpha}{11+2\alpha}\}$-GMMS allocation. For our case, this means that we cannot obtain a better than 0.404-GMMS approximation directly from Theorem 4.4.

Suppose now that we obtain an α-MMS allocation. Proposition 4.5 and Corollary 4.9 in [5] show that this cannot yield a constant factor approximation for either EFX or PMMS. The same then is true for GMMS. Similarly, Propositions 4.4 and 4.8 in [5] show that an α-PMMS allocation does not necessarily yield a constant approximation for EFX or MMS, and thus neither for GMMS. Hence, our Theorems 4.9 and 4.10 cannot provide the guarantees we have in Theorems 4.4, 4.6 and 4.7 for EFX and GMMS.

The only implications left to examine is when we start with an arbitrary α-GMMS allocation. Since GMMS allocations were not considered by Amanatidis et al. [5], we show here the following result.

Proposition Appendix B.1. *Let $\alpha \in (0,1)$. For $n \geq 2$, an α-GMMS allocation is not necessarily an $(\frac{\alpha}{2-\alpha} + \varepsilon)$-EF1 or an $(\alpha + \varepsilon)$-PMMS, or a β-EFX allocation for any $\varepsilon > 0, \beta \in (0,1)$.*

Proof. Consider the following instance with n agents and $4k + n$ goods for $k \geq 1$. Let $\beta \in (0,1)$. We focus on agent 1 and we have

$$
v_1(g_j) = \begin{cases}
\frac{\alpha}{2k} & 1 \leq j \leq 2k\,, \\
\frac{2-\alpha}{2k} & 2k+1 \leq j \leq 4k\,, \\
0.01 & 4k+1 \leq j \leq 4k+2\,, \\
2/\beta & j = 4k+3\,, \\
1 & 4k+3 \leq j \leq 4k+n\,.
\end{cases}
$$

Let $\mathcal{A} = (A_1, \ldots, A_n) = (\{g_1, \ldots, g_{2k}\}, \{g_{2k+1}, \ldots, g_{4k+1}\}, \{g_{4k+2}, g_{4k+3}\}, \{g_{4k+4}\}, \ldots, \{g_{4k+n}\})$ and assume that agents 2 through n are not envious, i.e., they value their corresponding bundles much higher than everything else. It is only a matter of simple calculations to see that according to agent 1, this is an α-GMMS allocation which is, however, only an $\frac{\alpha}{2-\alpha}$-EF1, an α-GMMS and a $\frac{\beta}{2}$-EFX allocation. $\qquad\square$

Plugging $\alpha = \frac{2}{\phi+2}$ in Proposition Appendix B.1, we get that an arbitrary 0.553-GMMS allocation is not necessarily a 0.383-EF1 or a 0.554-PMMS allocation, and has no guarantee whatsoever with respect to EFX. Hence, our Theorem 4.6 (or even Theorem 4.7 for that matter) does not imply the guarantees we obtain for EFX and PMMS in Theorems 4.4 and 4.9 respectively.

References

[1] Martin Aleksandrov and Toby Walsh. Greedy algorithms for fair division of mixed manna. *CoRR*, abs/1911.11005, 2019. URL http://arxiv.org/abs/1911.11005.

[2] Georgios Amanatidis, Georgios Birmpas, and Evangelos Markakis. On truthful mechanisms for maximin share allocations. In Subbarao Kambhampati, editor, *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, IJCAI 2016, New York, NY, USA, 9-15 July 2016*, pages 31–37. IJCAI/AAAI Press, 2016. URL http://www.ijcai.org/Abstract/16/012.

[3] Georgios Amanatidis, Georgios Birmpas, George Christodoulou, and Evangelos Markakis. Truthful allocation mechanisms without payments: Characterization and implications on fairness. In *Proceedings of the 2017 ACM Conference on Economics and Computation, EC '17, Cambridge, MA, USA, June 26-30, 2017*, pages 545–562. ACM, 2017. doi: 10.1145/3033274.3085147. URL https://doi.org/10.1145/3033274.3085147.

[4] Georgios Amanatidis, Evangelos Markakis, Afshin Nikzad, and Amin Saberi. Approximation algorithms for computing maximin share allocations. *ACM Trans. Algorithms*, 13(4):52:1–52:28, 2017. doi: 10.1145/3147173. URL https://doi.org/10.1145/3147173.

[5] Georgios Amanatidis, Georgios Birmpas, and Vangelis Markakis. Comparing approximate relaxations of envy-freeness. In Jérôme Lang, editor, *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI 2018, July 13-19, 2018, Stockholm, Sweden.*, pages 42–48. ijcai.org, 2018. doi: 10.24963/ijcai.2018/6. URL https://doi.org/10.24963/ijcai.2018/6.

[6] Georgios Amanatidis, Georgios Birmpas, Aris Filos-Ratsikas, Alexandros Hollender, and Alexandros A. Voudouris. Maximum Nash welfare and other stories about EFX. *CoRR*, abs/2001.09838, 2020. URL https://arxiv.org/abs/2001.09838.

[7] Siddharth Barman and Sanath Kumar Krishna Murthy. Approximation algorithms for maximin fair division. In *Proceedings of the 2017 ACM Conference on Economics and Computation, EC '17, Cambridge, MA, USA, June 26-30, 2017*, pages 647–664. ACM, 2017. doi: 10.1145/3033274.3085136. URL https://doi.org/10.1145/3033274.3085136.

[8] Siddharth Barman, Arpita Biswas, Sanath Kumar Krishna Murthy, and Yadati Narahari. Groupwise maximin fair allocation of indivisible goods. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), New Orleans, LA, USA, February 2-7, 2018*, pages 917–924. AAAI Press, 2018. URL https://www.aaai.org/ocs/index.php/AAAI/AAAI18/paper/view/16856.

[9] Siddharth Barman, Sanath Kumar Krishnamurthy, and Rohit Vaish. Finding fair and efficient allocations. In *Proceedings of the 2018 ACM Conference on Economics and Computation, Ithaca, NY, USA, June 18-22, 2018*, pages 557–574. ACM, 2018. doi: 10.1145/3219166.3219176. URL https://doi.org/10.1145/3219166.3219176.

[10] Steven J. Brams and Alan D. Taylor. *Fair division: From cake-cutting to dispute resolution*. Cambridge University Press, 1996. ISBN 978-0-521-55644-6.

[11] Félix Brandt, Vincent Conitzer, Ulle Endriss, Jérôme Lang, and Ariel D. Procaccia, editors. *Handbook of Computational Social Choice*. Cambridge University Press, 2016. ISBN 9781107446984. doi: 10.1017/CBO9781107446984. URL https://doi.org/10.1017/CBO9781107446984.

[12] Eric Budish. The combinatorial assignment problem: Approximate competitive equilibrium from equal incomes. *Journal of Political Economy*, 119(6):1061–1103, 2011.

[13] Ioannis Caragiannis, Nick Gravin, and Xin Huang. Envy-freeness up to any item with high Nash welfare: The virtue of donating items. In *Proceedings of the 2019 ACM Conference on Economics and Computation, EC 2019, Phoenix, AZ, USA, June 24-28, 2019.*, pages 527–545. ACM, 2019. doi: 10.1145/3328526.3329574. URL https://doi.org/10.1145/3328526.3329574.

[14] Ioannis Caragiannis, David Kurokawa, Hervé Moulin, Ariel D. Procaccia, Nisarg Shah, and Junxing Wang. The unreasonable fairness of maximum Nash welfare. *ACM Trans. Economics and Comput.*, 7(3):12:1–12:32, 2019. doi: 10.1145/3355902. URL https://doi.org/10.1145/3355902.

[15] Hau Chan, Jing Chen, Bo Li, and Xiaowei Wu. Maximin-aware allocations of indivisible goods. In Sarit Kraus, editor, *Proceedings of the Twenty-Eighth International Joint Conference on Artificial*

23

747   *Intelligence, IJCAI 2019, Macao, China, August 10-16, 2019*, pages 137–143. ijcai.org, 2019. doi: 10.24963/ijcai.2019/20. URL https://doi.org/10.24963/ijcai.2019/20.

[16] Bhaskar Ray Chaudhury, Jugal Garg, and Kurt Mehlhorn. EFX exists for three agents. *CoRR*, 750   abs/2002.05119, 2020. URL https://arxiv.org/abs/2002.05119.

[17] Bhaskar Ray Chaudhury, Telikepalli Kavitha, Kurt Mehlhorn, and Alkmini Sgouritsa. A little charity guarantees almost envy-freeness. In *Proceedings of the Fourty-First Annual ACM-SIAM* 753   *Symposium on Discrete Algorithms, SODA 2020, Salt Lake City, UT, USA, January 5-8, 2020.* SIAM, 2020. To appear.

[18] Duncan K. Foley. Resource allocation and the public sector. *Yale Economics Essays*, 7:45–98, 1967.
756 [19] George Gamow and Marvin Stern. *Puzzle-Math*. Viking press, 1958.

[20] Jugal Garg and Peter McGlaughlin. Improving Nash social welfare approximations. In Sarit Kraus, editor, *Proceedings of the Twenty-Eighth International Joint Conference on Artificial* 759   *Intelligence, IJCAI 2019, Macao, China, August 10-16, 2019*, pages 294–300. ijcai.org, 2019. doi: 10.24963/ijcai.2019/42. URL https://doi.org/10.24963/ijcai.2019/42.

[21] Jugal Garg and Setareh Taki. An improved approximation algorithm for maximin shares. *CoRR*, 762   abs/1903.00029, 2019. URL http://arxiv.org/abs/1903.00029.

[22] Jugal Garg, Peter McGlaughlin, and Setareh Taki. Approximating maximin share allocations. In Jeremy T. Fineman and Michael Mitzenmacher, editors, *2nd Symposium on Simplicity in Algorithms,* 765   *SOSA@SODA 2019, January 8-9, 2019 - San Diego, CA, USA*, volume 69 of *OASICS*, pages 20:1– 20:11. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2019. doi: 10.4230/OASIcs.SOSA.2019.20. URL https://doi.org/10.4230/OASIcs.SOSA.2019.20.

768 [23] Jugal Garg, Peter McGlaughlin, and Setareh Taki. Approximating maximin share allocations. In *2nd Symposium on Simplicity in Algorithms, SOSA@SODA 2019, January 8-9, 2019 - San Diego, CA, USA*, volume 69 of *OASICS*, pages 20:1–20:11. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 771   2019. doi: 10.4230/OASIcs.SOSA.2019.20. URL https://doi.org/10.4230/OASIcs.SOSA.2019.20.

[24] Mohammad Ghodsi, Mohammad Taghi Hajiaghayi, Masoud Seddighin, Saeed Seddighin, and Hadi Yami. Fair allocation of indivisible goods: Improvements and generalizations. In *Proceedings of the* 774   *2018 ACM Conference on Economics and Computation, Ithaca, NY, USA, June 18-22, 2018*, pages 539–556. ACM, 2018. doi: 10.1145/3219166.3219238. URL https://doi.org/10.1145/3219166.3219238.

777 [25] David Kurokawa. *Fair Division in Game Theoretic Settings*. PhD thesis, Carnegie Mellon University, 2017.

[26] David Kurokawa, Ariel D. Procaccia, and Junxing Wang. Fair enough: Guaranteeing approximate 780   maximin shares. *J. ACM*, 65(2):8:1–8:27, 2018. doi: 10.1145/3140756. URL https://doi.org/10.1145/3140756.

[27] Richard J. Lipton, Evangelos Markakis, Elchanan Mossel, and Amin Saberi. On approximately 783   fair allocations of indivisible goods. In *Proceedings 5th ACM Conference on Electronic Commerce (EC-2004), New York, NY, USA, May 17-20, 2004*, pages 125–131. ACM, 2004. doi: 10.1145/988772.988792. URL https://doi.org/10.1145/988772.988792.

786 [28] Evangelos Markakis. Approximation algorithms and hardness results for fair division with indivisible goods. In Ulle Endriss, editor, *Trends in Computational Social Choice*, chapter 12. AI Access, 2017.

[29] Hervé Moulin. Uniform externalities: Two axioms for fair allocation. *Journal of Public Economics*, 789   43(3):305–326, 1990.

[30] Hervé Moulin. *Fair division and collective welfare*. MIT Press, 2003.

[31] Benjamin Plaut and Tim Roughgarden. Almost envy-freeness with general valuations. In *Proceedings* 792   *of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2018, New Orleans, LA, USA, January 7-10, 2018*, pages 2584–2603. SIAM, 2018. doi: 10.1137/1.9781611975031.165. URL https://doi.org/10.1137/1.9781611975031.165.

795 [32] Jack M. Robertson and William A. Webb. *Cake-cutting algorithms: Be fair if you can*. A K Peters, 1998. ISBN 978-1-56881-076-8.

[33] Hugo Steinhaus. The problem of fair division. *Econometrica*, 16:101–104, 1948.

798 [34] Warut Suksompong. Approximate maximin shares for groups of agents. *Mathematical Social Sciences*, 92:40–47, 2018. doi: 10.1016/j.mathsocsci.2017.09.004. URL https://doi.org/10.1016/j.mathsocsci.2017.09.004.

801 [35] Hal R. Varian. Equity, envy and efficiency. *Journal of Economic Theory*, 9:63–91, 1974.