Achieving Envy-Freeness Through Items Sale

Vittorio Bilò
Department of Mathematics and Physics "Ennio De Giorgi", University of Salento, Italy

Evangelos Markakis
Department of Informatics, Athens University of Economics and Business, Greece
Input Output Global (IOG)

Cosimo Vinci
Department of Mathematics and Physics "Ennio De Giorgi", University of Salento, Italy

Abstract

We consider a fair division setting of allocating indivisible items to a set of agents. In order to cope with the well-known impossibility results related to the non-existence of envy-free allocations, we allow the option of selling some of the items so as to compensate envious agents with monetary rewards. In fact, this approach is not new in practice, as it is applied in some countries in inheritance or divorce cases. A drawback of this approach is that it may create a value loss, since the market value derived by selling an item can be less than the value perceived by the agents. Therefore, given the market values of all items, a natural goal is to identify which items to sell so as to arrive at an envy-free allocation, while at the same time maximizing the overall social welfare. Our work is focused on the algorithmic study of this problem, and we provide both positive and negative results on its approximability. When the agents have a commonly accepted value for each item, our results show a sharp separation between the cases of two or more agents. In particular, we establish a PTAS for two agents, and we complement this with a hardness result, that for three or more agents, the best approximation guarantee is provided by essentially selling all items. This hardness barrier, however, is relieved when the number of distinct item values is constant, as we provide an efficient algorithm for any number of agents. We also explore the generalization to heterogeneous valuations, where the hardness result continues to hold, and where we provide positive results for certain special cases.

2012 ACM Subject Classification Theory of computation → Approximation algorithms analysis; Theory of computation → Algorithmic game theory and mechanism design; Applied computing → Economics

Keywords and phrases Fair Item Allocation, Approximation Algorithms, Envy-freeness, Markets

Digital Object Identifier 10.4230/LIPIcs.ESA.2024.26

Funding This work is partially supported by: GNCS-INdAM - Programma Professori Visitatori; the PNRR MIUR project FAIR - Future AI Research (PE00000013), Spoke 9 - Green-aware AI; MUR - PNRR IF Agro@intesa; the Project SERICS (PE00000014) under the NRRP MUR program funded by the EU – NGEU; the Italian MIUR PRIN 2017 project ALGADIMAR - Algorithms, Games, and Digital Markets (2017R9FHSR_002); the framework of H.F.R.I call "Basic research Financing (Horizontal support of all Sciences)" under the National Recovery and Resilience Plan "Greece 2.0" funded by the European Union – NextGenerationEU (H.F.R.I. Project Number: 15877).

1 Introduction

Fair division refers to the algorithmic question of allocating resources or tasks to a set of agents according to some justice criteria. It is by now a prominent area within Algorithmic Game Theory and Computational Social Choice, [12, Part II], dating back to the origins of the civil society. One of the most natural and well studied notions of fairness is *envy-freeness* [18]: a division is envy-free if everyone thinks that her share is at least as valuable as the

32nd Annual European Symposium on Algorithms (ESA 2024).
Editors: Timothy Chan, Johannes Fischer, John Iacono, and Grzegorz Herman; Article No. 26; pp. 26:1–26:16
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

share of any other agent. In the presence of indivisible items however, obtaining an envy-free allocation is much more challenging [17], and it is well known that, in the majority of cases, envy-free divisions do not exist.

An approach that has been followed by several works, in order to cope with these existential issues, is to focus on relaxations of envy-freeness (for more on this we refer to our related work section). Another natural direction that comes into mind is to insist on envy-freeness but provide some compensation (e.g., monetary) to the agents who may feel unhappy by a proposed division. Such models have been considered in the literature, where money is either coming as an external subsidy from a third party or is already part of the initial endowment. Under this setting, [20] investigated the question of determining the minimum amount of money needed to obtain an envy-free division.

In this work, we also allow for monetary rewards, but we choose a different approach, as already initiated in [22]: we require that the money used to compensate the envious agents has to be raised from the set of available items, by selling some of them. This is what happens, for instance, in inheritance division. To provide some examples, as stated in Article n.9 of the New York Laws - Real Property Actions and Article n.720 of the Italian Civil Code, whenever an agreement is not possible, part of the inheritance can be sold through an auction. The same practice is also used in divorce settlements. Clearly, envy-freeness is then always feasible by selling, if needed, the whole inheritance, and equally sharing the proceeds. However, the amount of money raised by this process can be fairly below the real value of the sold items for at least two reasons. First, the bidders who participate in this type of auctions usually aim at winning items at very low prices; secondly, running an auction bears organizational costs which need to be subtracted from the proceeds. Thus, it is in the interest of the heirs to determine an envy-free division by selling assets with as little value loss as possible. This gives rise to an interesting optimization problem of determining which items to sell so as to arrive at an envy-free allocation, with optimal social welfare. Algorithmically, this question has been largely unexplored, with the exception of a particular case handled in [22].

1.1 Contribution

Assuming that we are given the *market value* of each item as input, i.e., the money that can be raised by selling it, we embark on a thorough investigation of algorithmic and complexity questions for our problem and provide an almost tight set of results.

We start in Section 3 with the case where all agents have the same value for each item. After establishing NP-hardness, which can be easily shown even for 2 agents, our main results exhibit a sharp separation on the approximability between the cases of $n = 2$ and $n \geq 3$ agents. In particular, we prove that, with at least three agents, no polynomial time algorithm can obtain a solution that performs better than the one which sells all items, unless P = NP. On the other hand, for two agents, we are able to design a polynomial time approximation scheme (PTAS), under the assumption that the market value of each item is not smaller than half of the common agents' value. The idea behind the PTAS is to enumerate all partial allocations of the most valuable items, whose number is a constant depending on the desired approximation guarantee. Each such partial allocation, which consists of the two bundles assigned to the agents together with the bundle of sold items, is then completed processing the remaining items by non-increasing value. At every step, the next item is allocated to the agent having the lower valued bundle, until we reach a situation where it is possible to equalize the two bundles by using the money raised from the already sold items and from selling a subset of the not-yet-processed ones. The main technical effort is needed to show that, if this condition occurs, then the final allocation can be made envy-free at the expense of a negligible loss of social welfare, while, if the condition never occurs, then it is not possible to obtain an envy-free solution from the starting partial allocation. Finally, our last result in Section 3 is the design of a dynamic programming algorithm which runs in polynomial time when the number of distinct item values is constant; this assumption is in line with several other recent works on fair-division, e.g., [5, 1].

In Section 4, we then move to the case where agents can have heterogeneous valuations. While all computational barriers from Section 3 carry over to this case as well, we are able to obtain two additional positive results. First, we focus on the setting where the values that an agent i has for the items lie in an interval of the form $[x_i, \beta x_i]$, where β is common across all agents. This means, essentially, that each agent attributes the same value to all items, up to a factor of β. For a constant number of agents, and for a constant value of β, we are able to design again a PTAS. This is very different from the PTAS of Section 3 and is based on an appropriate combination of two main ideas. First, by using a linear programming formulation, we compute a fractional solution with a bounded number of fractionally assigned items. Then, we apply a "reverse" version of the envy cycle elimination algorithm [24], so as to decide which items to sell, in addition to the fractional ones. We believe that this could be of independent interest for other allocation problems as well. Finally, at the end of Section 4, where we drop the assumption on β being constant, we also provide a pseudo-polynomial time algorithm.

Due to lack of space, all omitted proofs are deferred to the full version of this work.

1.2 Related Work

In terms of the model that we study, the work most related to ours is [22]. Their main focus however is not algorithmic but instead aims at comparing the *Price of Fairness* with and without selling items (defined as the ratio of the optimal social welfare versus the welfare attainable at an envy-free allocation). Their work also includes one algorithmic result, namely a PTAS, but for a case of only two heterogeneous agents, and under assumptions that are incomparable to our results of Section 4, where we consider any constant number of heterogeneous agents.

The use of monetary compensations, as a means to achieve envy-freeness with indivisible goods, has also been studied from various other angles in the literature, dating back e.g., to [27, 25]. The models that have been studied most often consider cases where each agent receives at most a single good (motivated by rent division) in addition to money. Given an a priori fixed amount of money, [2] yields an algorithm for determining an envy-free allocation. Improved algorithms were also provided in follow up works in [6, 23], and a more general model was considered in [9]. More recently, [20] take an optimization approach of minimizing the amount of required money for achieving envy-freeness (and without any restrictions in the items given to an agent). The main conceptual difference with our work is that in all these models, money is viewed as an already existing subsidy, whereas in our case it comes from selling some of the available items, which also leads to welfare loss.

Moving away from monetary rewards, there have been by now various other approaches for addressing the lack of envy-free allocations under indivisible items. The most popular one is the quest for relaxed notions of fairness, tailored for indivisible items. The notions of envy freeness up to 1 good (EF1) [13, 24], and envy freeness up to any good (EFX) [16, 19] are the two most representative examples, that have motivated a vast amount of recent works. An alternative direction was initiated in [14] where allocations satisfying the EFX criterion (and with high Nash welfare) were shown to exist when some items remain unallocated; giving birth to fairness *with charity*. For an overview of these notions and the relevant results, we refer the reader to the recent survey [3] and the references therein.

Finally, another relevant line of works concerns the quantification of welfare performance subject to fairness constraints. One way to formalize this is via the Price of Fairness [15], defined in the beginning of this section. The Price of Fairness w.r.t. other fairness criteria, such as relaxations of envy-freeness was later studied in [8], and further results with tight bounds were also given in [7, 21]. We refer again to the survey [3] for a more complete overview.

2 Definitions

We consider a set $[m] := \{1, \dots, m\}$ of m indivisible items to be allocated to a set $[n]$ of n agents. We assume that for every item j, there is a commonly accepted value $v(j)$, by all agents[1]. The vector $\boldsymbol{v} = (v(1), \dots, v(m))$ induces an additive valuation function $v : 2^{[m]} \mapsto \mathbb{N}$, so that for every subset $S \subseteq [m]$, the value of S is $v(S) = \sum_{j \in S} v(j)$.

An additional choice, instead of allocating all items to the agents, is to sell some of them in exchange of money. The rationale here is that if allocating all items cannot result in a fair allocation, we could use monetary compensations from sold items to achieve a more acceptable outcome. This may come at some value loss, since selling an item in the market can lead to a lower price than the value perceived by the agents. In particular, we assume that we are given a *market value* vector $\boldsymbol{v}_0 = (v_0(1), \dots, v_0(m))$, so that $v_0(j)$ is the monetary amount that can be obtained by selling item j, with $v_0(j) \leq v(j)$, for every $j \in [m]$. Viewing the vector $\boldsymbol{v}_0$ as inducing an (alternative) additive valuation function, we have that for every $S \subseteq [m]$, the money obtained by selling the items of S is equal to $v_0(S) = \sum_{j \in S} v_0(j)$.

Given an instance defined by a tuple $(n, m, \boldsymbol{v}, \boldsymbol{v}_0)$, an *allocation with items sale* is a partition of $[m]$ into $n+1$ subsets $\boldsymbol{X} = (X_0, X_1, \dots, X_n)$, such that, for each $i \in [n]$, X_i is the bundle allocated to agent i and X_0 is the set of items which are sold. Hence, the money made from $\boldsymbol{X}$ is $v_0(X_0)$. The *social welfare* of an allocation with items sale is given by $SW(\boldsymbol{X}) = v_0(X_0) + \sum_{i \in [n]} v(X_i)$. We will say that an allocation $\boldsymbol{X} = (X_0, X_1, \dots, X_n)$ is an *envy-free allocation with items sale* (from now on, simply EF-IS), if there exists a split of the money $v_0(X_0)$ into n amounts $\mu_1, \dots, \mu_n$, such that, for any two agents $i, i' \in [n]$, $v(X_i) + \mu_i \geq v(X_{i'}) + \mu_{i'}$. Since this needs to hold for any pair of agents, we can simplify the definition of EF-IS as follows. Define the maximum envy of an agent i under an allocation $\boldsymbol{X}$ as $e_i^{max}(\boldsymbol{X}) = \max_{i' \in [n]}\{v(X_{i'})\} - v(X_i)$ (note that, as i' can be also equal to i, $e_i^{max}(\boldsymbol{X}) \geq 0$). Then, an allocation is EF-IS if and only if

$$v_0(X_0) \geq \sum_{i \in [n]} e_i^{max}(\boldsymbol{X}). \quad (1)$$

Hence, whenever the above equation holds, it means that there is enough money to compensate all agents having non-zero maximum envy.

We observe that an EF-IS allocation always exists: simply sell all items and share equally all the money. We call this allocation, the *basic* EF-IS allocation. Therefore, this gives rise to the natural optimization problem of finding the best EF-IS allocation in terms of social welfare. This constitutes the focus of our work, and we define it formally below.

[1] We start with this modeling choice, as it is common in inheritance or divorce settlements, that items such as land properties or cars have a common value to the agents. In Section 4, we explore extensions beyond this assumption.

BEST-EF-IS

Given an instance $(n, m, \boldsymbol{v}, \boldsymbol{v}_0)$ on n agents, m items, value vector $\boldsymbol{v}$ and market value vector $\boldsymbol{v}_0$, find an allocation $\boldsymbol{X} = (X_0, X_1, \ldots, X_n)$ that is EF-IS and attains maximum social welfare.

3 Hardness and Approximability

We begin with defining a parameter that plays a fundamental role in the majority of our results. Given an instance $(n, m, \boldsymbol{v}, \boldsymbol{v}_0)$, let $\alpha := \min_{j\in[m]:v(j)>0}\left\{\frac{v_0(j)}{v(j)}\right\} \in [0, 1]$, be the largest discrepancy between the market value and the commonly accepted value of any item. We observe the following.

▶ **Observation 1.** *The basic EF-IS allocation is an α-approximation of BEST-EF-IS.*

3.1 Hardness Results

We address first two extreme cases of the problem. By Observation 1, BEST-EF-IS is trivial when $\alpha = 1$. On the opposite side, when $\alpha = 0$, the problem cannot be approximated up to any finite factor, even for $n = 2$.

▶ **Theorem 2.** *For $n = 2$ and $\alpha = 0$, BEST-EF-IS cannot be approximated up to any finite factor, unless P = NP.*

For the more interesting cases of $\alpha \in (0, 1)$, a direct reduction from Partition yields NP-hardness even for $n = 2$.

▶ **Theorem 3.** *For $n = 2$ and $\alpha \in (0, 1)$, BEST-EF-IS is NP-hard.*

The next theorem is our main result from this subsection. It implies that the α-approximate solution achieved by the basic EF-IS allocation is the best we can hope for, when we have at least 3 agents.

▶ **Theorem 4.** *For any $n \geq 3$ and $\alpha \in (0, 1)$, BEST-EF-IS cannot be approximated with a ratio better than $\alpha + \epsilon$, for any constant $\epsilon > 0$, unless P = NP.*

Proof. Fix n, α and ϵ and let $s(\alpha)$ be the number of bits needed to encode α. We show the claim by a gap producing reduction from Partition. Consider an instance of Partition I made up of $p > \max\left\{n, s(\alpha), \frac{n}{(n-2)(1-\alpha)}, \frac{2(1-\alpha)}{\epsilon n}\right\}$ positive integers $w_1, \ldots, w_p$, such that $\sum_{j\in[p]} w_j = 2B$. The Partition problem asks to find a subset $A \subseteq [p]$ such that $\sum_{j\in A} w_j = B$. Create an instance I' of BEST-EF-IS with $m = p + n$ items such that $v(j) = w_j$ for each $j \in [p]$, $v(p + 1) = v(p + 2) = (p - 1)B$, and $v(j) = pB$ for each $j > p + 2$. We call any item of value pB a *big item*, any item of value $(p-1)B$ an *almost big item* and the remaining items *small items*. Note that there are $n - 2$ big items and that $\sum_{j\in[m]} v(j) = npB$. The vector $\boldsymbol{v}_0$ is defined in such a way that $v_0(j) = \alpha v(j)$ for each $j \in [m]$. Note that, by the choice of p, the representation of I' is polynomial in that of I, for any value of n, α and ϵ. The remaining proof is then completed by establishing the following lemma:

▶ **Lemma 5.** *If I admits a partition, we can construct in polynomial time an EF-IS allocation $\boldsymbol{X}$, with $SW(\boldsymbol{X}) = npB$. Conversely, if I does not admit a partition, any EF-IS allocation $\boldsymbol{X}$ satisfies $SW(\boldsymbol{X}) < (\alpha + \epsilon)npB$.* ◀

Algorithm 1 COMPLETE$(m, \boldsymbol{v}, \boldsymbol{v}_0, q, X_0, X_1, X_2)$.

Input: an ordered instance $I = (m, \boldsymbol{v}, \boldsymbol{v}_0)$, an integer $q < m$ and an EF-IS allocation (X_0, X_1, X_2) restricted to the first q items of I
Output: an EF-IS allocation for I

$Y_1 \leftarrow \emptyset$, $Y_2 \leftarrow \emptyset$
if $q = m - 1$ **then**
    $Y_0 \leftarrow \{m\}$
else
    $Y_0 \leftarrow \{q+1, q+2\}$
end if
for $j \leftarrow q + 3$ to m **do**
    if $v(Y_1) \le v(Y_2)$ **then**
        $Y_1 \leftarrow Y_1 \cup \{j\}$
    else
        $Y_2 \leftarrow Y_2 \cup \{j\}$
    end if
end for
return $(X_0 \cup Y_0, X_1 \cup Y_1, X_2 \cup Y_2)$

Finally, in the following theorem we show that, if the number of agents is not fixed, the hardness of approximation holds even if the item values are polynomially bounded in m and n.

▶ **Theorem 6.** *For any $\alpha \in (0, 1)$ and $\epsilon > 0$, approximating BEST-EF-IS to better than $\alpha + \epsilon$ is strongly NP-hard.*

3.2 A PTAS for Two Agents when $\alpha \ge 1/2$

In light of the hardness results shown in the previous subsection, non-trivial approximation algorithms, without any further assumptions, are possible only for the case of $n = 2$ and $\alpha \in (0, 1)$. This is the focus of this subsection, and our main result is the design of a polynomial time approximation scheme (PTAS), under the mild assumption that $\alpha \ge 1/2$ (the market value is never less than 50% of the real value for any item). We feel that this assumption is not far from what we would expect in practice. If the items (e.g. in an inheritance or divorce case) had a way too low market value, it would not even make sense to sell them at all.

Within this subsection, we use $(m, \boldsymbol{v}, \boldsymbol{v}_0)$ to describe an instance, since $n = 2$. Note that the condition described in Equation (1) for an allocation $\boldsymbol{X} = (X_0, X_1, X_2)$ to be EF-IS simplifies to $v_0(X_0) \ge |v(X_1) - v(X_2)|$. Before illustrating our PTAS, we start with a simple procedure, called COMPLETE, that will be used as a subroutine of our algorithm. Say that an instance is *ordered* if items are sorted by non-increasing value. COMPLETE works as follows: given an ordered instance $I = (m, \boldsymbol{v}, \boldsymbol{v}_0)$, an integer $q < m$ and an EF-IS allocation (X_0, X_1, X_2) restricted to the first q items of I, it sells items $q + 1$ and $q + 2$ (if any), and then, for each $j \ge q + 3$, item j is assigned to the agent whose bundle, considering only the items allocated by COMPLETE so far, has the smaller total value, breaking ties in favour of agent 1.

As shown in the lemma below, COMPLETE helps us in extending an EF-IS allocation of an initial subset of items, to an EF-IS allocation over all the items.

Algorithm 2 SolveBestEF-IS$(m, \boldsymbol{v}, \boldsymbol{v}_0)$.

Input: an instance $I = (m, \boldsymbol{v}, \boldsymbol{v}_0)$, with $\alpha \geq 1/2$, and a value $\epsilon > 0$

Output: a $(1-\epsilon)$-approximate solution

1: sort the items by non-increasing value
2: $\overline{q} \leftarrow \min\{m, \lceil 5(1-\alpha-\epsilon)/\epsilon \rceil\}$
3: $\boldsymbol{S} \leftarrow (\emptyset, \emptyset, \emptyset)$, $maxWelf \leftarrow 0$
4: **for** any allocation with items sale $\boldsymbol{X} := (X_0, X_1, X_2)$ restricted to the first $\overline{q}$ items **do**
5: $\quad \boldsymbol{Y} := (Y_0, Y_1, Y_2) \leftarrow$Extend$(I, \overline{q}, \boldsymbol{X})$
6: $\quad$ **if** $SW(\boldsymbol{Y}) > maxWelf$ **then**
7: $\quad\quad \boldsymbol{S} \leftarrow \boldsymbol{Y}$, $maxWelf \leftarrow SW(\boldsymbol{Y})$
8: $\quad$ **end if**
9: **end for**
10: **return** $\boldsymbol{S}$

▶ **Lemma 7.** *Given an ordered instance $I = (m, \boldsymbol{v}, \boldsymbol{v}_0)$, an integer $q < m$ and an EF-IS allocation $\boldsymbol{X} = (X_0, X_1, X_2)$ restricted to the first q items of I, Complete$(I, q, \boldsymbol{X})$ returns in $O(m)$ time an EF-IS allocation for I selling items that are worth a total value of at most $v(X_0) + 2v(q+1)$.*

For any $\epsilon > 0$, our algorithm, called SolveBestEF-IS, returns a $(1-\epsilon)$-approximation for BEST-EF-IS. The algorithm relies on an initial brute force enumeration of all possible allocations with items sale, restricted to the first $\overline{q}$ items, where $\overline{q}$ is a constant that depends on ϵ. As each item $j \in [\overline{q}]$ can be either assigned to one of the two agents or sold (three possible choices), there is a total of $3^{\overline{q}}$ possible outcomes. For each outcome, corresponding to a partial allocation with items sale $\boldsymbol{X} = (X_0, X_1, X_2)$ restricted to the first $\overline{q}$ items, function Extend is invoked. This function takes an ordered instance $I = (m, \boldsymbol{v}, \boldsymbol{v}_0)$, an integer $q < m$ and an allocation $\boldsymbol{X} = (X_0, X_1, X_2)$ restricted to the first q items of I. Running it with $q = \overline{q}$, it checks whether $\boldsymbol{X}$ can be extended to the remaining $m - \overline{q}$ items, that is, *without altering the allocation of the first $\overline{q}$ items*, so as to yield an EF-IS allocation for I. If this is possible, we are able to show that Extend returns an EF-IS allocation which, besides the items in X_0, it also sells items that are worth a total value of at most $5v(\overline{q}+1)$. If this is not possible, we show that the starting guessed allocation $\boldsymbol{X}$ cannot be extended to an EF-IS allocation, and so Extend returns the basic EF-IS allocation. Upon enumeration of all possible allocations with items sale restricted to the first $\overline{q}$ items, SolveBestEF-IS returns the EF-IS allocation with the largest social welfare.

The core of SolveBestEF-IS is the function Extend, based on a careful analysis of the various cases that may arise. We provide here an overview of how it works and refer the reader to full version for further details.

Lines 1–6

The first part of the algorithm checks whether $\boldsymbol{X}$ is an EF-IS allocation restricted to the first $\overline{q}$ items of I. If this is the case, we invoke Complete to obtain an EF-IS allocation for I. In the negative case, if no items are left to extend $\boldsymbol{X}$ (i.e., $\overline{q} = m$), we return the basic EF-IS allocation; otherwise, we conclude that $v_0(X_0) < |v(X_1) - v(X_2)|$ and define $\overline{X}$ as the set of the remaining $m - q$ items.

Lines 7–13

To proceed, line 7 (possibly) swaps X_1 and X_2 so as to have $v(X_1) > v(X_2) + v_0(X_0)$. The function now checks whether the items in $\overline{X}$ can be used to obtain an EF-IS allocation out of $\boldsymbol{X}$. If $v(X_1) > v(X_2) + v_0(X_0) + v(\overline{X})$, this is not possible and the basic EF-IS allocation is returned (lines 8–10). Otherwise, $v(X_1) \leq v(X_2) + v_0(X_0) + v(\overline{X})$. If this holds at equality, then $(X_0, X_1, X_2 \cup \overline{X})$ is an EF-IS allocation and is returned (lines 11–13).

Lines 15–27

We now have $v(X_1) < v(X_2) + v_0(X_0) + v(\overline{X})$. The first goal here is to check if selling a single item from $\overline{X}$ suffices. Thus, in the while-loop at lines 16–23, the function considers sequentially all items whose removal from $\overline{X}$ invalidates inequality $v(X_1) < v(X_2) + v_0(X_0) + v(\overline{X})$. If, by selling item j, enough money can be raised to cover the difference between $v(X_1)$ and $v(X_2) + v_0(X_0) + v(\overline{X} \setminus \{j\})$, then the EF-IS allocation $(X_0 \cup \{j\}, X_1, X_2 \cup \overline{X} \setminus \{j\})$ is returned (line 18). Otherwise, item j is added to a special set of items S (line 20). If the while-loop terminates without returning a solution, we shall prove that the only way to possibly extend the guessed allocation $\boldsymbol{X}$ to an EF-IS allocation for I is to assign the items of S to X_2. Lines 25–27 check whether the new partial allocation after the addition of S to X_2 is EF-IS restricted to the first $q + |S|$ items. If so, Complete is invoked to obtain an EF-IS allocation for I.

Lines 29–31

If the execution arrives at line 29, it must be either $v(X_1) > v(X_2) + v_0(X_0)$ or $v(X_2) > v(X_1) + v_0(X_0)$. In the latter case, we are essentially in the same situation as at the beginning of the function, except for the fact that $\boldsymbol{X}$ has been extended to the first $q + |S|$ items. Observe that, since we started with $v(X_1) > v(X_2) + v_0(X_0)$ and arrived at a situation in which $v(X_2) > v(X_1) + v_0(X_0)$, there was some progress in between, i.e., $S \neq \emptyset$). Thus, the function sets $q = q + |S|$ and restarts from line 1.

Lines 33–44

This is the final phase of the algorithm. If we arrive at line 33, we have $v(X_1) > v(X_2) + v_0(X_0)$ and $v(X_1) < v(X_2) + v_0(X_0) + v(\overline{X})$. The while-loop at lines 33–35 keeps adding items to the second bundle and stops as soon as an item j is added to X_2, such that $v(X_1) \leq v(X_2) + v_0(X_0)$. If this condition holds at equality, we can again invoke Complete to obtain an EF-IS allocation for I (lines 36–38). If the execution arrives at line 41, we have $v(X_1) < v(X_2) + v_0(X_0)$ and $v(X_1) \geq v(X_2 \setminus \{j\}) + v_0(X_0)$. At lines 41–43, the function computes a set of items Y to be sold guaranteeing that the allocation $(X_0 \cup Y, X_1, X_2 \setminus \{j\})$ is EF-IS. We prove that such a set Y does exist, and by invoking Complete, the algorithm terminates with an EF-IS allocation for I (line 44).

The following lemma shows the correctness and complexity of function Extend.

▶ **Lemma 8.** *For any input $(I = (m, \boldsymbol{v}, \boldsymbol{v}_0), q, X_0, X_1, X_2)$, Extend returns an EF-IS allocation for I in $O(m)$ time. If there exists an EF-IS allocation for I allocating the first q items as in (X_0, X_1, X_2), then an EF-IS allocation selling items that are worth at most $v(X_0) + 5v(q + 1)$ is returned; otherwise, the basic EF-IS allocation is returned.*

To prove Lemma 8, we need first some auxiliary results that establish some basic properties of the whole algorithm.

Algorithm 3 EXTEND$(m, \boldsymbol{v}, \boldsymbol{v}_0, q, X_0, X_1, X_2)$.

Input: an ordered instance $I = (m, \boldsymbol{v}, \boldsymbol{v}_0)$ with $\alpha \geq 1/2$, a positive integer $q < m$ and an allocation $\boldsymbol{X} = (X_0, X_1, X_2)$ restricted to the first q items of I
Output: an EF-IS allocation for I

1: **if** $v_0(X_0) \geq |v(X_1) - v(X_2)|$ **then**
2:     **return** COMPLETE$(I, q, \boldsymbol{X})$
3: **else if** $q = m$ **then**
4:     **return** $([m], \emptyset, \emptyset)$
5: **end if**
6: $\overline{X} \leftarrow [m] \setminus (X_0 \cup X_1 \cup X_2)$
7: swap X_1 and X_2 so that $v(X_1) > v(X_2) + v_0(X_0)$
8: **if** $v(X_1) > v(X_2) + v_0(X_0) + v(\overline{X})$ **then**
9:     **return** $([m], \emptyset, \emptyset)$
10: **end if**
11: **if** $v(X_1) = v(X_2) + v_0(X_0) + v(\overline{X})$ **then**
12:     **return** $(X_0, X_1, X_2 \cup \overline{X})$
13: **end if**
14: % if we reach this point, then $v(X_1) > v(X_2) + v_0(X_0)$ and $v(X_1) < v(X_2) + v_0(X_0) + v(\overline{X})$
15: $j \leftarrow q + 1$, $S \leftarrow \emptyset$
16: **while** $j \leq m$ && $v(j) \geq v(X_2) + v_0(X_0) + v(\overline{X}) - v(X_1)$ **do**
17:     **if** $v_0(j) \geq v(X_1) - v(X_2) - v_0(X_0) - v(\overline{X}) + v(j)$ **then**
18:         **return** $(X_0 \cup \{j\}, X_1, X_2 \cup \overline{X} \setminus \{j\})$
19:     **else**
20:         $S \leftarrow S \cup \{j\}$
21:     **end if**
22:     $j \leftarrow j + 1$
23: **end while**
24: $X_2 \leftarrow X_2 \cup S$, $\overline{X} \leftarrow \overline{X} \setminus S$
25: **if** $v_0(X_0) \geq |v(X_1) - v(X_2)|$ **then**
26:     **return** COMPLETE$(I, q + |S|, \boldsymbol{X})$
27: **end if**
28: % if we reach this point, then either $v(X_1) > v(X_2) + v_0(X_0)$ or $v(X_2) > v(X_1) + v_0(X_0)$
29: **if** $v(X_2) > v(X_1) + v_0(X_0)$ **then**
30:     set $q \leftarrow q + |S|$ and **goto** line 1
31: **end if**
32: % if we reach this point, $v(X_1) > v(X_2) + v_0(X_0)$ and, by line 14 and the fact that in between items are only moved from $\overline{X}$ to X_2, $v(X_1) < v(X_2) + v_0(X_0) + v(\overline{X})$
33: **while** $(v(X_1) > v(X_2) + v_0(X_0))$ **do**
34:     $X_2 \leftarrow X_2 \cup \{j\}$, $j \leftarrow j + 1$
35: **end while**
36: **if** $(v(X_1) = v(X_2) + v_0(X_0))$ **then**
37:     **return** COMPLETE$(I, j - 1, \boldsymbol{X})$
38: **end if**
39: $Y \leftarrow \{j - 1\}$, $X_2 \leftarrow X_2 \setminus \{j - 1\}$
40: % if we reach this point, then $v(X_1) < v(X_2) + v_0(X_0)$ and $v(X_1) \geq v(X_2 \setminus \{j\}) + v_0(X_0)$
41: **while** $(v_0(Y) < v(X_1) - v(X_2) - v_0(X_0))$ **do**
42:     $Y \leftarrow Y \cup \{j\}$, $j \leftarrow j + 1$
43: **end while**
44: **return** COMPLETE$(I, j - 1, X_0 \cup Y, X_1, X_2)$

▶ **Lemma 9.** *Suppose* EXTEND *reaches line 24, after constructing the set of items S, via line 20. If there exists an EF-IS allocation for I allocating the first q items of $[m]$ as in (X_0, X_1, X_2), then it allocates the first $q + |S|$ items of $[m]$ as in $(X_0, X_1, X_2 \cup S)$.*

▶ **Lemma 10.** *The while-loop at lines 33–35 always terminates; moreover, the while-loop at lines 41–43 always terminates returning a set Y such that $v(Y) \leq 3v(q+1)$.*

Proof of Lemma 8. First of all, by Lemma 10, we are guaranteed that Extend always terminates. Moreover, each returned allocation is always EF-IS either by definition, or by inspection combined with Lemma 7. Regarding the complexity, it is not difficult to see that Extend can be executed in $O(m)$ time. In fact, the first 13 lines of the function, apart from basic, constant-time operations, require the computation of quantities such as $\overline{X}$, $X_2 \cup \overline{X}$, $v(X_0)$, $v(X_1)$, $v(X_2)$, $v(\overline{X})$, taking $O(m)$ time, and the invocation of Complete which, by Lemma 7, needs $O(m)$ time. From line 14 onwards, the function essentially processes all items sequentially and, for each processed item, constant-time operations are performed. The only non-constant-time operation is the invocation to Complete which requires $O(m)$ and is performed only once during any execution of Extend.

Now, let us bound the value of sold items in any returned solution other than the basic one. The solution returned at line 2 sells items that are worth a total value of at most $v(X_0) + 2v(q + 1)$, due to Lemma 7; the one returned at line 12 sells items that are worth a total value of $v(X_0)$; that returned at line 18, sells items that are worth a total value of $v(X_0) + v(j) \leq v(X_0) + v(q + 1)$; those returned at lines 26 and 37, sell items that are worth in total at most $v(X_0) + 2v(q + 1)$, by Lemma 7; finally, the one returned at line 44 sells items that are worth a total value of at most $v(X_0) + 5v(q + 1)$, by Lemmas 7 and 10.

We are left to show that Extend returns the basic EF-IS allocation only if no EF-IS allocation extending $\boldsymbol{X} = (X_0, X_1, X_2)$ exists. Extend returns the basic EF-IS allocation only at lines 4 and 9. Observe that, when this happens, there is no chance of getting an EF-IS allocation from $\boldsymbol{X}$. This partial allocation, in fact, is either given in input, or obtained from the input after possible repeated additions of the special set S of items to X_2, at line 24. By Lemma 9, these additions do not prevent the extension of the current allocation to an EF-IS one. So, whenever we reach the point that Extend has to return the basic EF-IS allocation, it is because it has arrived at an allocation which cannot be extended to become EF-IS and such an allocation has been obtained by performing unavoidable choices only. ◀

Putting everything together, we can show our main result.

▶ **Theorem 11.** *There is a PTAS for BEST-EF-IS with two agents when $\alpha \geq 1/2$.*

Proof. Fix an instance $I = (m, \boldsymbol{v}, \boldsymbol{v}_0)$ of BEST-EF-IS. By Lemma 8, we get that, for any $0 < \epsilon < 1 - \alpha$, SolveBestEF-IS returns an EF-IS allocation in $O(3^{\overline{q}} m + m \log m)$ time, with $\overline{q} = O(1/\epsilon)$. To show the approximation guarantee, let $\boldsymbol{O}$ be the optimal allocation of the problem, and let (X_0^*, X_1^*, X_2^*) be the partial allocation corresponding to $\boldsymbol{O}$ and restricted to the first $\overline{q}$ items. Set $\overline{X}^* := [m] \setminus (X_0^* \cup X_1^* \cup X_2^*)$. When SolveBestEF-IS calls Extend with input I, $\overline{q}$ and (X_0^*, X_1^*, X_2^*), it receives an EF-IS allocation $\boldsymbol{S}$ where, additionally to X_0^*, a set of items $Z \subseteq \overline{X}^*$ is sold and such that $v(Z) \leq 5v(\overline{q}+1)$, by Lemma 8. We derive $SW(\boldsymbol{S}) = v(X_1^* \cup X_2^*) + v_0(X_0^*) + v_0(Z) + v(\overline{X}^* \setminus Z)$. On the other hand, $SW(\boldsymbol{O}) \leq v(X_1^* \cup X_2^*) + v_0(X_0^*) + v(Z) + v(\overline{X}^* \setminus Z)$. So, the approximation guarantee achieved by $\boldsymbol{S}$ is at least

$$\frac{v(X_1^* \cup X_2^*) + v_0(X_0^*) + v_0(Z) + v(\overline{X}^* \setminus Z)}{v(X_1^* \cup X_2^*) + v_0(X_0^*) + v(Z) + v(\overline{X}^* \setminus Z)} \geq \frac{v(X_1^* \cup X_2^*) + v_0(X_0^*) + \alpha v(Z) + v(\overline{X}^* \setminus Z)}{v(X_1^* \cup X_2^*) + v_0(X_0^*) + v(Z) + v(\overline{X}^* \setminus Z)}.$$

This value is minimized when the common terms are as small as possible, while $v(Z)$ is as large as possible. This is achieved when $v(Z) = 5v(\overline{q}+1)$, $X_1^* = X_2^* = \overline{X}^* \setminus Z = \emptyset$ and $X_0^* = [\overline{q}]$, with $v(X_0^*) = \overline{q}v(\overline{q}+1)$. We derive that the approximation guarantee achieved by

$\boldsymbol{S}$ is at least $\frac{\overline{q}+5\alpha}{\overline{q}+5} \geq 1-\epsilon$, for each $\overline{q} \geq \frac{5(1-\alpha-\epsilon)}{\epsilon}$. As SolveBestEF-IS returns the EF-IS allocation with the highest social welfare, among the ones returned by Extend, the claim follows. ◀

▶ Remark 12. At a first glance, it seems that the value of α plays a major role only within function COMPLETE. This function can be easily extended to any $\alpha \leq 1/k$, with k being an integer such that $k > 2$, by selling the next k items, rather than simply the next two ones. However, the fact that $\alpha \geq 1/2$, is fundamental to prove that the while loop at lines 41–43 of function EXTEND always terminates (see second part of the claim of Lemma 10). So, in order to extend the PTAS below the threshold 1/2, additional arguments need to be elaborated.

3.3 An Exact Algorithm for a Small Number of Distinct Values

In this subsection, we consider the case in which there is a small number of distinct item values. In particular, we assume that there are T distinct item values, say $w_1 < w_2 < \ldots < w_T$, and that, for any $s \in [T]$, there are m_s items of value w_s. Obviously, it must hold that $\sum_{s\in[T]} m_s = m$. We design a dynamic programming algorithm that solves BEST-EF-IS in polynomial-time when T is a constant.

▶ **Theorem 13.** *Let T be the number of distinct item values. BEST-EF-IS can be solved in time $O(n(m/T)^{2T}T)$.*

4 Extensions to Heterogeneous Agents

In this section, we consider a generalization of our model to the case where the items may not have the same value for all agents. The most natural extension is the one in which each agent i has her own additive valuation function v_i, so that $v_i(j)$ is the value of agent i for item j and $\boldsymbol{v}_i = (v_i(j))_{j\in[m]}$ denotes the vector of all item values for agent i. Under heterogeneous valuation functions, we need to be more careful about the market value vector $\boldsymbol{v}_0$. As also done in [22], we assume that for every item j, the market value satisfies $v_0(j) \leq min_i v_i(j)$. We view this as a minimal assumption, that should hold so that no agent can have more value by selling an item rather than by owning it. Furthermore, we assume w.l.o.g. that the considered allocation problems do not contain any dummy item j with $v_i(j) = 0$ for all $i \in [n]$.

The following result, due to [20] but recast in our framework, shows how to determine in polynomial time if a given allocation with items sale is EF-IS.

▶ **Lemma 14** ([20]). *Given an allocation problem $(n, m, (\boldsymbol{v}_i)_{i\in[n]}, \boldsymbol{v}_0)$ with heterogeneous valuations, let $\boldsymbol{X} = (X_0, X_1, \ldots, X_n)$ be a an allocation with items sale of $[m]$. One can check in $O(mn + n^3)$ time if $\boldsymbol{X}$ is EF-IS.*

Let $\alpha := \min_{i\in[n],j\in[m]:v_i(j)>0} \left\{\frac{v_0(j)}{v_i(j)}\right\} \in [0,1]$ be the parameter defined similarly as in Section 3. As before, the basic EF-IS allocation is a feasible solution and trivially constitutes an optimal one when $\alpha = 1$. Moreover, the hardness results of Section 3.1 continue to hold under heterogeneous valuations. Therefore the problem is NP-hard, and with 3 agents or more, there is no approximation factor better than α. Nevertheless, we are still able to provide some positive results under certain assumptions. In particular, let $\beta := \max_{i\in[n]} \frac{\max_{j\in[m]} v_i(j)}{\min_{j\in[m]:v_i(j)>0} v_i(j)}$, denote the maximum ratio between the highest and the lowest (non-zero) valuable item of any agent. We obtain below a PTAS, if n, β and $1/\alpha$ are bounded by a constant. This can be seen as generalizing the PTAS of Section 3.2, even beyond the two agent case, but only with a constant β. The technique that we use however is quite different from the PTAS of Section 3.2. Furthermore, we also obtain an exact pseudo-polynomial time algorithm for the case in which the number of agents is constant, without any further assumption.

4.1 A PTAS for Few Heterogeneous Agents with $\beta = O(1)$ and $\alpha = \Omega(1)$

Within this section we assume that n, β and $1/\alpha$ are bounded by some constant. Moreover, given two integers k, n, we denote by $[n]_k$ the set $\{k, k+1, \dots, n\}$, if $k \leq n$ and the empty set otherwise.

In Algorithm 4, we provide the pseudo-code of our PTAS, which we call Cut&Sell. We provide here a brief overview and intuition of how it works. For a fixed $\epsilon > 0$, we parameterize the analysis with a constant, but sufficiently large integer q (defined in Algorithm 4), with $q = O(1/\epsilon)$. If $m \leq q$ the algorithm computes, via a polynomial-time brute-force search, an EF-IS allocation $\boldsymbol{X}$ that maximizes the social welfare. Otherwise, when $m > q$, it executes the following procedures:

Cut Procedure

This procedure computes a fractional allocation $\boldsymbol{z} = (z_{i,j})_{i\in[n],j\in[m]}$, with $z_{i,j}$ denoting the fraction of j assigned to i, that maximizes the social welfare, is envy-free, and cuts at most $n^2 - n + 1$ items into fractional pieces. To do this in polynomial time, one can resort to linear programming. In particular, we have that the optimal solution of the following linear program in variables $(z_{i,j})_{i\in[n],j\in[m]}$ corresponds to a social welfare maximizing fractional allocation, subject to envy-freeness:

$$
\begin{aligned}
\max \;\; & SW(\boldsymbol{z}) := \sum_{i\in[n],j\in[m]} v_{i,j} z_{i,j} \\
s.t. \; & \sum_{j\in[m]} v_{i,j} z_{i,j} \geq \sum_{j\in[m]} v_{i,j} z_{h,j}, \;\; \forall i, h \in [n], i \neq h \\
& \sum_{i\in[n]} z_{i,j} = 1, \;\; \forall j \in [m] \\
& z_{i,j} \geq 0, \;\; \forall i \in [n], j \in [m].
\end{aligned} \tag{2}
$$

By Lemma 15 below, we can efficiently find the desired optimal solution of linear program (2).

▶ **Lemma 15.** *An optimal solution $\boldsymbol{z} = (z_{i,j})_{i\in[n],j\in[m]}$ of linear program (2) that satisfies $|\{j \in [m] : \exists i \in [n], 0 < z_{i,j} < 1\}| \leq n^2 - n + 1$ can be computed in polynomial time.*

Sell Procedure

Let X_0 be the set of items that are assigned fractionally in $\boldsymbol{z}$. We first sell X_0 and derive the (integral) allocation with items sale $\boldsymbol{X} = (X_0, X_1, \dots, X_n)$, where each j in X_i is assigned integrally to i in $\boldsymbol{z}$. Now, given the allocation $\boldsymbol{X'} = (X_1, \dots, X_n)$ restricted to the unsold items of $\boldsymbol{X}$, the *envy-graph* of $\boldsymbol{X'}$ is a graph $G_{\boldsymbol{X'}} = (V, E_{\boldsymbol{X'}})$ having $V = [n]$, and containing an edge $(i, h) \in [n]^2$ if and only if agent i is envious of h (that is, $v_i(X_i) < v_i(X_h)$). The algorithm permutes the bundles in such a way that the resulting envy-graph of the allocation restricted to unsold items is acyclic. This step is implemented by a sub-routine called

EnvyCycleElimination introduced in [24] (we refer the reader to such work for a detailed description of the sub-routine). Then, the algorithm alternates among the following two steps, until the obtained allocation $\boldsymbol{X}$ is EF-IS: (i) It picks a sink $i \in [n]$ in the resulting (acyclic) envy-graph, i.e., an agent that is not currently envious of anyone. Then, the algorithm removes and sells an arbitrary good j from the bundle of i (i.e., j is added to X_0). (ii) It applies again EnvyCycleElimination, so that the resulting envy-graph of the allocation restricted to unsold items becomes acyclic.

We shall prove that at the end, the algorithm terminates with an EF-IS allocation and with the desired approximation.

▶ **Theorem 16.** *There is a PTAS for BEST-EF-IS with n heterogeneous agents, when n, β and $1/\alpha$ are all $O(1)$.*

Proof Sketch. We first argue about the complexity of the algorithm. If $m \leq q$, Cut&Sell enumerates $(n+1)^m \leq (n+1)^q = O(1)$ distinct allocations with item sales; for each of them, it verifies in polynomial time if it is EF-IS (by Lemma 14), and finally returns the one of maximum welfare. The case $m > q$ is polynomial as it requires to find an optimal fractional allocation via linear programming and applies at most $O(m)$ times the envy-cycle-elimination procedure.

The technically more involved part is to show that Cut&Sell guarantees a $(1-\epsilon)$-approximation. If $m \leq q$, an optimal solution is returned by enumeration. If $m > q$, Cut&Sell first computes an optimal fractional envy-free-solution $\boldsymbol{z}$, whose social welfare $SW(\boldsymbol{z})$ is used as an optimality benchmark for BEST-EF-IS. Then, by using the hypothesis that $n, 1/\alpha, \beta$ are bounded by a constant, we show that the items with positive value added to X_0 during the Sell procedure, before reaching an EF-IS allocation, have low value compared to $SW(\boldsymbol{z})$. This fact is used to show that the allocation returned by the algorithm is a $(1-\epsilon)$-approximation. ◀

4.2 A Pseudo-Polynomial Time Algorithm for Few Heterogeneous Agents

Our final result is that for a constant number of agents, BEST-EF-IS admits a pseudo-polynomial time algorithm. Considering the hardness result provided in Theorem 6, a limitation on the number of agents is necessary to obtain a pseudo-polynomial time algorithm, unless P = NP.

▶ **Theorem 17.** *BEST-EF-IS can be solved in $O(mn^2V^{n^2})$ time for heterogeneous valuations and in $O(mnV^n)$ time for identical ones, where $V = \max_{i\in[n]}\{v_i([m])\}$ denotes the maximum value for the entire set of items.*

5 Conclusions and Future work

Our work explores from an algorithmic perspective the model of fair division of indivisible items initiated in [22], and provides an almost complete picture on its status. This model considers the possibility of selling items in order to compensate envious agents in a proposed allocation.

Despite the large amount of research work devoted in the last years to the study of relaxed notions of envy-freeness, the approach of items sale has remained largely unexplored. This may look strange since, although relaxed notions of envy-freeness such as EFX and

Algorithm 4 $\text{Cut\&Sell}(m, \boldsymbol{v}, \boldsymbol{v}_0)$.

Input: $\epsilon > 0$ and an instance $I = (m, \boldsymbol{v}, \boldsymbol{v}_0)$ with heterogeneous agents, $\alpha \in (0,1)$, $\beta \geq 1$ and $v_i(j) > 0$ for any $i \in [n]$, $j \in [m]$

Output: an EF-IS allocation for I

$q \leftarrow \lceil n(n^2+1)\beta\,(\beta-\alpha)\,/(\epsilon\alpha^2)\rceil$
if $m \leq q$ **then**
    Find the best EF-IS allocation by enumeration
else
    % Cut Procedure
    $\boldsymbol{z} = (z_{i,j})_{i\in[n],j\in[m]} \leftarrow$ an optimal envy-free fractional allocation, in which at most $n^2 - n + 1$ items are cut into two or more fractional pieces.
    % Sell Procedure
    $X_0 \leftarrow \{j \in [m] : \exists i, \text{ s.t. } z_{i,j} \in (0,1), i \in [n]\}$
    for $i = 1, \dots, n$ **do**
        $X_i \leftarrow \{j \in [m] : z_{i,j} = 1\}$
    end for
    $\boldsymbol{X} = (X_0, \dots, X_n)$, $\boldsymbol{X}' \leftarrow (X_1, \dots, X_n)$
    $\boldsymbol{X}' \leftarrow \text{EnvyCycleElimination}(\boldsymbol{X}')$
    while $\boldsymbol{X}$ is not EF-IS **do**
        Let $i \in [n]$ such that $v_i(X_i) \geq v_i(X_h)$ for each $h \in [n]$ (break ties arbitrarily)
        Let $j \in X_i$ (break ties arbitrarily)
        $X_i \leftarrow X_i \setminus \{j\}$, $X_0 \leftarrow X_0 \cup \{j\}$
        $\boldsymbol{X}' \leftarrow (X_1, \dots, X_n)$
        $\boldsymbol{X}' \leftarrow \text{EnvyCycleElimination}(\boldsymbol{X}')$
    end while
end if
return $\boldsymbol{X}$

EF1 provide theoretically interesting and elegant solutions to the non-existence of envy-free allocations, from a practical point of view there are many cases in which these solutions are highly unfair (think, for instance, of the famous basic case of a high-valued item and two agents). A possible reason for this under-consideration might come from the intrinsic difficulty of the problem, as witnessed by the strong computational barriers we proved in Subsection 3.1. However, we have also shown that, under some (in some cases even mild) assumptions, interesting positive results are possible.

An interesting open question that arises is whether we can extend the existence of a PTAS for two agents, in the case of $\alpha \in (0, 1/2)$ and identical valuations, and also in the case of arbitrary α and heterogeneous valuations (without further assumptions on other parameters). Furthermore, it would be nice to study the effects of items sale for other variants of fair allocation problems, such as for other notions of fairness (e.g., proportionality, EFX or maximin shares) or for more general valuations beyond additivity, or for problems with additional constraints (e.g., under connectivity constraints [10, 11, 26]). Finally, it would be interesting to study the case of strategic agents, as in [4], who may misreport their valuations to increase their utility.

References

1 Hannaneh Akrami, Bhaskar Ray Chaudhury, Martin Hoefer, Kurt Mehlhorn, Marco Schmalhofer, Golnoosh Shahkarami, Giovanna Varricchio, Quentin Vermande, and Ernest van Wijland. Maximizing nash social welfare in 2-value instances. In *Thirty-Sixth AAAI Conference on Artificial Intelligence, AAAI 2022*, pages 4760–4767, 2022.

2 A. Alkan, G. Demange, and D. Gale. Fair allocation of indivisible goods and criteria of justice. *Econometrica*, 59(4):1023–1039, 1991.

3 Georgios Amanatidis, Haris Aziz, Georgios Birmpas, Aris Filos-Ratsikas, Bo Li, Hervé Moulin, Alexandros A. Voudouris, and Xiaowei Wu. Fair division of indivisible goods: Recent progress and open questions. *Artif. Intell.*, 322:103965, 2023.

4 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*, pages 545–562. ACM, 2017.

5 Georgios Amanatidis, Georgios Birmpas, Aris Filos-Ratsikas, Alexandros Hollender, and Alexandros A. Voudouris. Maximum nash welfare and other stories about EFX. *Theor. Comput. Sci.*, 863:69–85, 2021.

6 E. Aragones. A derivation of the money Rawlsian solution. *Social Choice and Welfare*, 12(3):267–276, 1995.

7 Siddharth Barman, Umang Bhaskar, and Nisarg Shah. Optimal bounds on the price of fairness for indivisible goods. In *Proceedings of the 16th International Conference on Web and Internet Economics, WINE 2020*, pages 356–369, 2020.

8 Xiaohui Bei, Xinhang Lu, Pasin Manurangsi, and Warut Suksompong. The price of fairness for indivisible goods. *Theory Comput. Syst.*, 65(7):1069–1093, 2021.

9 C. Bevia. Fair allocation in a general model with indivisible goods. *Review of Economic Design*, 3:195–213, 1998.

10 Vittorio Bilò, Ioannis Caragiannis, Michele Flammini, Ayumi Igarashi, Gianpiero Monaco, Dominik Peters, Cosimo Vinci, and William S. Zwicker. Almost envy-free allocations with connected bundles. *Games Econ. Behav.*, 131:197–221, 2022.

11 Sylvain Bouveret, Katarína Cechlárová, Edith Elkind, Ayumi Igarashi, and Dominik Peters. Fair division of a graph. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI 2017*, pages 135–141, 2017.

12 F. Brandt, V. Conitzer, U. Endriss, J. Lang, and A. D. Procaccia, editors. *Handbook of Computational Social Choice*. Cambridge University Press, 2016.

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

14 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*, pages 527–545. ACM, 2019.

15 Ioannis Caragiannis, Christos Kaklamanis, Panagiotis Kanellopoulos, and Maria Kyropoulou. The efficiency of fair division. *Theory Comput. Syst.*, 50(4):589–610, 2012.

16 Ioannis Caragiannis, David Kurokawa, Hervé Moulin, Ariel D. Procaccia, Nisarg Shah, and Junxing Wang. The unreasonable fairness of maximum nash welfare. In Vincent Conitzer, Dirk Bergemann, and Yiling Chen, editors, *Proceedings of the 2016 ACM Conference on Economics and Computation, EC 2016*, pages 305–322. ACM, 2016.

17 D. Foley. Resource allocation and the public sector. *Yale Econ Essays*, 7(1):45–98, 1967.

18 G. Gamow and M. Stern, editors. *Puzzle-math*. Viking Press, 1958.

19 Laurent Gourvès, Jérôme Monnot, and Lydia Tlilane. Near fairness in matroids. In *Proceedings of the 21st European Conference on Artificial Intelligence, ECAI 2014*, pages 393–398, 2014.

20 D. Halpern and N. Shah. Fair division with subsidy. In *Proceedings of the 12th International Symposium on Algorithmic Game Theory (SAGT)*, LNCS 11801, pages 374–389. Springer, 2019.

21 Daniel Halpern and Nisarg Shah. Fair and efficient resource allocation with partial information. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI 2021*, pages 224–230, 2021.

22 Jeremy Karp, Aleksandr M. Kazachkov, and Ariel D. Procaccia. Envy-free division of sellable goods. In *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence, AAAI 2014*, pages 728–734, 2014.

23 F. Klijn. An algorithm for envy-free allocations in an economy with indivisible objects and money. *Social Choice and Welfare*, 17(2):201–215, 2000.

24 Richard J Lipton, Evangelos Markakis, Elchanan Mossel, and Amin Saberi. On approximately fair allocations of indivisible goods. In *Proceedings of the 5th ACM Conference on Electronic Commerce (EC) 2004*, pages 125–131, 2004.

25 E. Maskin. On the fair allocation of indivisible goods. In G. R. Feiwel, editor, *Arrow and the Foundations of the Theory of Economic Policy*, pages 341–349. Palgrave Macmillan, 1987.

26 Warut Suksompong. Fairly allocating contiguous blocks of indivisible items. *Discret. Appl. Math.*, 260:227–236, 2019.

27 L. Svensson. Large indivisibles: An analysis with respect to price equilibrium and fairness. *Econometrica*, 51(4):939–954, 1983.