



# Computing Fair and Efficient Allocations with Few Utility Values

Jugal Garg and Aniket Murhekar<sup>(✉)</sup>

University of Illinois at Urbana-Champaign, Champaign, USA  
{jugal, aniket2}@illinois.edu

**Abstract.** We study the problem of allocating indivisible goods among agents with additive valuations in a fair and efficient manner, when agents have *few* utility values for the goods. We consider the compelling fairness notion of envy-freeness up to any good (EFX) in conjunction with Pareto-optimality (PO). Amanatidis et al. [1] showed that when there are at most two utility values, an EFX allocation can be computed in polynomial-time. We improve this result by showing that for such instances an allocation that is EFX and PO can be computed in polynomial-time. This is the first class apart from identical or binary valuations, for which EFX+PO allocations are shown to exist and are polynomial-time computable. In contrast, we show that when there are three utility values, EFX+PO allocations need not exist, and even deciding if EFX+PO allocations exist is NP-hard.

Our techniques allow us to obtain similar results for the fairness notion of equitability up to any good (EQX) together with PO. We show that for instances with two positive values an EQX+PO allocation can be computed in polynomial-time, and deciding if an EQX+PO allocation exists is NP-hard when there are three utility values.

We also study the problem of maximizing Nash welfare (MNW), and show that our EFX+PO algorithm returns an allocation that approximates the MNW to a factor of 1.061 for two valued instances, in addition to being EFX+PO. In contrast, we show that for three valued instances, computing an MNW allocation is APX-hard.

**Keywords:** Fair and efficient allocation · EFX · Nash welfare · EQX

## 1 Introduction

The problem of fair division was formally introduced by Steinhaus [36], and has since been extensively studied in various fields, including economics and computer science [10, 32]. It concerns allocating resources (goods) to agents in a *fair* and *efficient* manner, and has various practical applications such as rent division, division of inheritance, course allocation, and government auctions.

---

Supported by NSF Grant CCF-1942321 (CAREER).

© Springer Nature Switzerland AG 2021

I. Caragiannis and K. A. Hansen (Eds.): SAGT 2021, LNCS 12885, pp. 345–359, 2021.

[https://doi.org/10.1007/978-3-030-85947-3\\_23](https://doi.org/10.1007/978-3-030-85947-3_23)

Arguably, the most popular notion of fairness is *envy-freeness* (EF) [19,37], which requires that every agent prefers their own bundle of goods to that of any other. However in the case of *indivisible* goods, EF allocations need not even exist (consider allocating 1 good among 2 agents). This motivated the study of its relaxations. One such relaxation is *envy-freeness up to one good* (EF1) allocation, defined by Budish [11], where every agent prefers their own bundle to the bundle of any other agent after removing *some* good from the other agent's bundle. It is well-known that an EF1 allocation always exists and is polynomial-time computable [29]. However, an EF1 allocation may be unsatisfactory because it allows the removal of the *most valuable* good from the other agent's bundle, which might be the main reason for huge envy to exist in the first place. Therefore, stronger fairness notions are desirable in many settings.

A stronger notion is called *envy-free up to any good* (EFX), defined by Caragiannis et al. [12], which requires every agent to prefer their bundle over the bundle of any other agent after removing *any* good from the other agent's bundle. Clearly, any allocation that is EFX is also EF1, but not vice-versa. The existence of EFX allocations is known for identical valuations [34], and was recently shown for 3 agents with additive valuations [15].<sup>1</sup> At the same time, we want the output allocation to be efficient because a fair allocation by itself may be highly inefficient. Consider for example two agents  $A_1$  and  $A_2$  and 2 goods  $g_1$  and  $g_2$  where  $A_i$  values only  $g_i$  and does not value the other good. The allocation in which  $g_1$  is assigned to  $A_2$  and  $g_2$  is assigned to  $A_1$  is clearly EFX. However both agents get zero utility, which is highly inefficient. The allocation in which  $g_i$  is assigned to  $A_i$  is more desirable since it is both fair as well as efficient.

The standard notion of economic efficiency is Pareto optimality (PO). An allocation is said to be PO if no other allocation makes an agent better off without making someone else worse off. A stronger notion called fractional Pareto optimality (fPO) requires that no other fractional allocation makes an agent better off without making someone else worse off. Every fPO allocation is therefore PO, but not vice-versa (see the appendix for an example). Another reason to prefer fPO allocations over PO allocations is that the former admit efficient verification while the latter do not: given an allocation, it can be checked in polynomial time if it is fPO [5], whereas checking if an allocation is PO is coNP-complete [27]. Hence if a centralized entity responsible for allocating resources claims the allocation is fPO, each agent can individually verify that this is indeed the case; in contrast such a check is not efficiently possible if the guarantee is only PO.

An important question is whether the notions of fairness (EF1 or EFX) can be achieved in conjunction with the efficiency notions (PO or fPO). Further, if yes, then whether they can be computed in polynomial-time. For this, the concept of *Nash welfare* provides a partial answer. The Nash welfare is defined as the geometric mean of the agents' utilities, and by maximizing it we achieve

---

<sup>1</sup> Settling the (non-)existence of EFX allocations is considered the biggest open question in fair division [35]; see [16] and references therein for recent progress on this problem.

a tradeoff between efficiency and fairness. Caragiannis et al. [12] showed that the maximum Nash welfare (MNW) allocations are EF1 and PO under additive valuations. However, the problem of computing an MNW allocation is APX-hard [28] (hard to approximate). Bypassing this barrier, Barman et al. [5] devised a pseudo-polynomial-time algorithm that computes an EF1+PO allocation. In a recent paper, Garg *et al.* [23] showed that an EF1+fPO allocation can be computed in pseudo-polynomial time. For the special case of binary additive valuations an MNW allocation is EFX+fPO, and is known to be polynomial-time computable [6, 18].

## 1.1 Our Contributions

In this work, we obtain several novel results on the notions of EFX, EQX, PO, and MNW, especially for instances in which agents have few values for the goods. A fair division instance is called  $k$ -valued if values that agents have for the goods belong a set of size  $k$ .

**EFX.** Recently, Amanatidis *et al.* [1] showed that for 2-valued instances any MNW allocation is EFX+PO, but left open the question of whether it can be computed in polynomial-time. They presented a polynomial-time algorithm which computes an EFX allocation for 2-valued instances, however, the outcome of their algorithm need not be PO (see the appendix for an example). In this work, we show EFX+fPO allocations always exist for 2-valued instances and can be computed in polynomial-time.<sup>2</sup> Further, apart from the classes of identical valuations and binary valuations, this is the first class for which EFX+PO allocations exist and can be computed in polynomial-time.

In general, EFX+PO allocations are not guaranteed to exist [34]. We therefore ask the natural question: what is the complexity of *checking* if an instance admits an EFX+PO allocation? We show that this problem is NP-hard, somewhat surprisingly, even for 3-valued instances.

**EQX.** Our techniques allow us to obtain similar results for the fairness notion of *equitability up to any good* [20, 26]. An allocation is said to be EQX (resp. EQ1) if the utility an agent gets from her bundle is no less than the utility any other agent gets after removing *any* (resp. some) good from their bundle. We show that for positive 2-valued instances, an EQX+PO allocation can be computed in polynomial-time, and in contrast, even checking existence of EQX+PO allocations for 3-valued instances is NP-hard.

**MNW.** Our EFX+PO algorithm returns an allocation that approximates the maximum Nash welfare to a factor of 1.061 in addition to being EFX and PO. This guarantee is better than the best known 1.45-approximation algorithm of [5] for the MNW problem.

Amanatidis *et al.* [1] showed that computing an MNW allocation is NP-hard for 3-values instances, which, as they remark “extends the hardness aspect, but

<sup>2</sup> Our results extend to the much broader class where there are two values  $\{a_i, b_i\}$  per agent, but  $a_i/b_i$  is the same across agents.

not the inapproximability, of the result of Lee [28] for 5-valued instances”, who had shown that MNW is NP-hard to approximate within a factor of 1.00008. In our work, we extend the inapproximability aspect too, and show that it is NP-hard to approximate the MNW to a factor of 1.00019, even for 3-valued instances, which is better than Lee’s result.

Thus, for the problems of computing (i) EFX+PO, (ii) EQX+PO, and (iii) MNW allocations, our work improves the state-of-the-art and also crucially pin-points the boundary between tractable and intractable cases.

## 1.2 Other Related Work

Barman *et al.* [5] showed that for  $n$  agents and  $m$  goods, an EF1+PO allocation can be computed in time  $\text{poly}(n, m, v_{max})$ , where  $v_{max}$  is the maximum utility value. Their algorithm first perturbs the values to a desirable form, and then computes an EF1+fPO allocation for the perturbed instance, which for a small-enough perturbation is EF1+PO for the original instance. Their approach is via *integral market-equilibria*, which guarantees fPO at every step. Our algorithm uses a similar approach, with one main difference being that we do not need to consider any approximate instance and can work directly with the given values. The outcome of our algorithm is EFX+fPO, which beats the guarantee of EF1+PO.

Another key difference is the *run-time analysis*: our arguments show termination in  $\text{poly}(n, m)$  time for 2-valued instances, even when  $v_{max} = 2^{\Omega(n+m)}$ , whereas the analysis of Barman *et al.* only shows a  $\text{poly}(n, m, v_{max})$  time bound, even for 2-valued instances.

Recently, Garg and Murhekar [33] showed that an EF1+fPO allocation can be computed in  $\text{poly}(n, m, v_{max})$ -time, by using integral market-equilibria. They also showed that an EF1+fPO allocation can be computed in  $\text{poly}(n, m)$ -time for  $k$ -valued instances where  $k$  is a constant, however they do not show that the allocation returned by their algorithm is EFX for 2-valued instances.

Freeman *et al.* [20] showed that EQ.1+PO allocations can be computed in pseudo-polynomial-time for instances with positive values. They also show that the leximin solution, i.e., the allocation that maximizes the minimum utility, and subject to this, maximizes the second minimum utility, and so on; is EQX+PO. However, as remarked in [34], computing a leximin solution is intractable.

Barman *et al.* [6] showed that for identical valuations, any EFX allocation provides a 1.061-approximation to the MNW. Garg *et al.* [21] show a 1.069-hardness of approximating MNW, although for 4-valued instances.

Instances with few values have been widely considered in the fair division literature: for instance Golovin [25] presents approximation algorithms and hardness results for computing max-min fair allocations in 3-valued instances; Aziz *et al.* [3] show PO is efficiently verifiable for 2-valued instances and coNP-hard for 3-valued instances; Aziz [2], and Vazirani and Yannakakis [38] study the Hylland-Zeckhauser scheme for probabilistic assignment of goods in 2-valued instances; Bogomolnaia and Moulin [9] study matching problems with 2-valued (dichotomous) preferences; Bliem *et al.* [8] study fixed-parameter tractability

for computing EF+PO allocations with parameter  $n + z$ , where  $z$  is the number of values; and Garg *et al.* [24] study leximin assignments of papers ranked by reviewers on a small scale, in particular they present an efficient algorithm for 2 ranks, i.e., “high or low interest” and show NP-hardness for 3 ranks. More generally, such instances have been studied in resource allocation contexts, including makespan minimization with 2 or 3 job sizes [13,39].

## 2 Preliminaries

For  $t \in \mathbb{N}$ , let  $[t]$  denote  $\{1, \dots, t\}$ .

**Problem Setting.** A fair division instance is a tuple  $(N, M, V)$ , where  $N = [n]$  is a set of  $n \in \mathbb{N}$  agents,  $M = [m]$  is the set of  $m \in \mathbb{N}$  indivisible items, and  $V = \{v_1, \dots, v_n\}$  is a set of utility functions, one for each agent  $i \in N$ . Each utility function  $v_i : M \rightarrow \mathbb{Z}_{\geq 0}$  is specified by  $m$  numbers  $v_{ij} \in \mathbb{Z}_{\geq 0}$ , one for each good  $j \in M$ , which denotes the value agent  $i$  has for good  $j$ . We assume that the valuation functions are additive, that is, for every agent  $i \in N$ , and for  $S \subseteq M$ ,  $v_i(S) = \sum_{j \in S} v_{ij}$ . Further, we assume that for every good  $j$ , there is some agent  $i$  such that  $v_{ij} > 0$ . Note that we can in general work with rational values without loss of generality, since they can be scaled to make them integral, and the efficiency and fairness guarantees we consider are scale-invariant.<sup>3</sup>

We call a fair division instance  $(N, M, V)$  a  $t$ -valued instance if  $|\{v_{ij} : i \in N, j \in M\}| = t$ . The class of 2-valued instances is made up of two disjoint fragments: binary instances, where all values  $v_{ij} \in \{0, 1\}$ ; and  $\{a, b\}$ -instances, where all values  $v_{ij} \in \{a, b\}$  for  $a, b \in \mathbb{Z}_{>0}$ . An important subclass of 3-valued instances is the  $\{0, a, b\}$  class, wherein all values  $v_{ij} \in \{0, a, b\}$  for  $a, b \in \mathbb{Z}_{>0}$ .

**Allocation.** An (integral) allocation  $\mathbf{x}$  of goods to agents is a  $n$ -partition  $(\mathbf{x}_1, \dots, \mathbf{x}_n)$  of the goods, where  $\mathbf{x}_i \subseteq M$  is the bundle of goods allotted to agent  $i$ , who gets a total value of  $v_i(\mathbf{x}_i)$ . A fractional allocation  $\mathbf{x} \in [0, 1]^{n \times m}$  is a fractional assignment of the goods to agents such that for each good  $j \in M$ ,  $\sum_{i \in N} x_{ij} = 1$ . Here,  $x_{ij} \in [0, 1]$  denotes the fraction of good  $j$  allotted to agent  $i$ . In a fractional allocation  $\mathbf{x}$ , an agent  $i$  receives a value of  $v_i(\mathbf{x}_i) = \sum_{j \in M} v_{ij}x_{ij}$ .

**Fairness Notions.** An allocation  $\mathbf{x}$  is said to be:

1. *Envy-free up to one good* (EF1) if for all  $i, h \in N$ , there exists a good  $j \in \mathbf{x}_h$  s.t.  $v_i(\mathbf{x}_i) \geq v_i(\mathbf{x}_h \setminus \{j\})$ .
2. *Envy-free up to any good* (EFX) if for all  $i, h \in N$  and for all goods  $j \in \mathbf{x}_h$  we have  $v_i(\mathbf{x}_i) \geq v_i(\mathbf{x}_h \setminus \{j\})$ .
3. *Equitable up to one good* (EQ.1) if for all  $i, h \in N$ , there exists a good  $j \in \mathbf{x}_h$  s.t.  $v_i(\mathbf{x}_i) \geq v_h(\mathbf{x}_h \setminus \{j\})$ .
4. *Equitable up to any good* (EQX) if for all  $i, h \in N$  and for all goods  $j \in \mathbf{x}_h$  we have  $v_i(\mathbf{x}_i) \geq v_h(\mathbf{x}_h \setminus \{j\})$ .

<sup>3</sup> The properties of EFX, PO, and Nash welfare are invariant under scaling, while EQX is not scale-invariant in general. However, in our algorithms this is not an issue since we only uniformly scale the valuations of all agents, which preserves EQX.

**Pareto-optimality.** An allocation  $\mathbf{y}$  dominates an allocation  $\mathbf{x}$  if for all  $i \in N$ ,  $v_i(\mathbf{y}_i) \geq v_i(\mathbf{x}_i)$  and there exists  $h \in N$  s.t.  $v_h(\mathbf{y}_h) > v_h(\mathbf{x}_h)$ . An allocation is said to be *Pareto-optimal* (PO) if no allocation dominates it. Further, an allocation is said to be *fractionally Pareto-optimal* (fPO) if no fractional allocation dominates it. Thus, any fPO allocation is PO, but not vice-versa (see the appendix for an example).

**Nash Welfare.** The Nash welfare of an allocation  $\mathbf{x}$  is given by  $NW(\mathbf{x}) = (\prod_{i \in N} v_i(\mathbf{x}_i))^{1/n}$ . An allocation that maximizes the NW is called an MNW allocation or Nash optimal allocation. An allocation  $\mathbf{x}$  *approximates* the MNW to a factor  $\alpha$  if  $\alpha \cdot NW(\mathbf{x}) \geq NW(\mathbf{x}^*)$ , where  $\mathbf{x}^*$  is an MNW allocation.

**Fisher Markets.** A Fisher market or a *market instance* is a tuple  $(N, M, V, e)$ , where  $N = [n]$  is a set of  $n \in \mathbb{N}$  agents,  $M = [m]$  is a set of  $m \in \mathbb{N}$  divisible goods,  $V = \{v_1, \dots, v_n\}$  is a set of additive (linear) utility functions, and  $e = \{e_1, \dots, e_n\}$  is the set of agents' budgets, where each  $e_i \geq 0$ . In this model, agents can fractionally share goods. Each agent aims to obtain a bundle of goods that maximizes her total value subject to her budget constraint.

A *market outcome* is a fractional allocation  $\mathbf{x}$  of the goods to the agents and a set of prices  $\mathbf{p} = (p_1, \dots, p_m)$  of the goods, where  $p_j \geq 0$  for every  $j \in M$ . The spending of an agent  $i$  under the market outcome  $(\mathbf{x}, \mathbf{p})$  is given by  $\mathbf{p}(\mathbf{x}_i) = \sum_{j \in M} p_j x_{ij}$ . For an agent  $i$ , we define the *bang-per-buck* ratio  $\alpha_{ij}$  of good  $j$  as  $v_{ij}/p_j$ , and the *maximum bang-per-buck* (MBB) ratio  $\alpha_i = \max_j \alpha_{ij}$ . We define  $m_{bb_i} = \{j \in M : \alpha_i = v_{ij}/p_j\}$ , called the *MBB-set*, to be the set of goods that give MBB to agent  $i$  at prices  $\mathbf{p}$ . A market outcome  $(\mathbf{x}, \mathbf{p})$  is said to be '*on MBB*' if for all agents  $i$  and goods  $j$ ,  $x_{ij} > 0 \Rightarrow j \in m_{bb_i}$ . For integral  $\mathbf{x}$ , this means  $\mathbf{x}_i \subseteq m_{bb_i}$ .

A market outcome  $(\mathbf{x}, \mathbf{p})$  is said to be a *market equilibrium* if (i) the market clears, i.e., all goods are fully allocated. Thus, for all  $j$ ,  $\sum_{i \in N} x_{ij} = 1$ , (ii) budget of all agents is exhausted, for all  $i \in N$ ,  $\sum_{j \in M} x_{ij} p_j = e_i$ , and (iii) agents only spend money on MBB goods, i.e.,  $(\mathbf{x}, \mathbf{p})$  is on MBB.

Market equilibria are an important tool in computing fair and efficient allocations because of their remarkable fairness and efficiency properties; see e.g., [4, 5, 14, 17, 20, 22, 31]. The First Welfare Theorem [30] shows that for a market equilibrium  $(\mathbf{x}, \mathbf{p})$  of a Fisher market instance  $\mathcal{M}$ , the allocation  $\mathbf{x}$  is fPO. We include a proof in the appendix for completeness.

**Theorem 1.** (*First Welfare Theorem [30]*) *Let  $(\mathbf{x}, \mathbf{p})$  be a equilibrium of a Fisher market  $\mathcal{M}$ . Then  $\mathbf{x}$  is fractionally Pareto-optimal.*

Given an allocation  $\mathbf{x}$  for a fair division instance  $(N, M, V)$  and a vector of prices  $\mathbf{p}$  for the goods such that  $(\mathbf{x}, \mathbf{p})$  is on MBB, one can define an associated Fisher market instance  $\mathcal{M} = (N, M, V, e)$  by setting  $e_i = \mathbf{p}(\mathbf{x}_i)$ . It is easy to see that  $(\mathbf{x}, \mathbf{p})$  is a market equilibrium of  $\mathcal{M}$ . Hence Theorem 1 implies:

**Corollary 1.** *Given a market outcome  $(\mathbf{x}, \mathbf{p})$  on MBB, the allocation  $\mathbf{x}$  is fPO.*

### 3 Computing EFX+PO Allocations

We first study the problem of computing an EFX+PO allocation for  $t$ -valued instances when  $t \in \{2, 3\}$ . We show that EFX+PO allocations can be computed in polynomial-time for 2-valued instances, and in contrast, computing such allocations for 3-valued instances is NP-hard.

#### 3.1 EFX+PO Allocations for 2-Valued Instances

We consider  $\{a, b\}$ -instances, as it is known EFX+PO allocations can be efficiently computed for binary instances. We remark that while the allocation returned by the algorithm Match&Freeze of Amanatidis *et al.*, [1] for  $\{a, b\}$ -instances is EFX, it need not be PO (example in appendix). We improve this result by showing that:

**Theorem 2.** *Given a fair division  $\{a, b\}$ -instance  $I = (N, M, V)$ , an allocation that is EFX, fPO and approximates the maximum Nash welfare to a factor of 1.061 can be computed in polynomial-time.*

We prove this theorem by showing that Algorithm 1 computes such an allocation. We first define some relevant terms, including the concept of price envy-freeness introduced by Barman *et al.* [5]. A market outcome  $(\mathbf{x}, \mathbf{p})$  is said to be *price envy-free up to one good* (pEF1) if for all agents  $i, h \in N$  there is a good  $j \in \mathbf{x}_h$  such that  $\mathbf{p}(\mathbf{x}_i) \geq \mathbf{p}(\mathbf{x}_h \setminus \{j\})$ . Similarly, we say it is pEFX if for all agents  $i, h \in N$ , and for all goods  $j \in \mathbf{x}_h$  it holds that  $\mathbf{p}(\mathbf{x}_i) \geq \mathbf{p}(\mathbf{x}_h \setminus \{j\})$ . For market outcomes on MBB, the pEFX condition implies the EFX condition:

**Lemma 1.** *Let  $(\mathbf{x}, \mathbf{p})$  be an integral market outcome on MBB. Then  $\mathbf{x}$  is fPO. If  $(\mathbf{x}, \mathbf{p})$  is pEFX, then  $\mathbf{x}$  is EFX.*

*Proof.* The fact that  $\mathbf{x}$  is fPO follows from Corollary 1. Since  $(\mathbf{x}, \mathbf{p})$  is pEFX, for all pairs of agents  $i, h \in N$ , and all goods  $j \in \mathbf{x}_h$  it holds that  $\mathbf{p}(\mathbf{x}_i) \geq \mathbf{p}(\mathbf{x}_h \setminus \{j\})$ . Since  $(\mathbf{x}, \mathbf{p})$  is on MBB,  $\mathbf{x}_i \subseteq mbb_i$ . Let  $\alpha_i$  be the MBB-ratio of  $i$  at the prices  $\mathbf{p}$ . By definition of MBB,  $v_i(\mathbf{x}_i) = \alpha_i \mathbf{p}(\mathbf{x}_i)$ , and  $v_i(\mathbf{x}_h \setminus \{j\}) \leq \alpha_i \mathbf{p}(\mathbf{x}_h \setminus \{j\})$ , for every  $j \in \mathbf{x}_h$ . Combining these, we get that  $\mathbf{x}$  is EFX.  $\square$

Given a price vector  $\mathbf{p}$ , we define the MBB graph to be the bipartite graph  $G = (N, M, E)$  where for an agent  $i$  and good  $j$ ,  $(i, j) \in E$  iff  $j \in mbb_i$ . Such edges are called *MBB edges*. Given an accompanying allocation  $\mathbf{x}$ , we supplement  $G$  to include *allocation edges*, an edge between agent  $i$  and good  $j$  if  $j \in \mathbf{x}_i$ .

We call the agent  $i$  with minimum  $\mathbf{p}(\mathbf{x}_i)$  a *least spender* (LS), where ties are broken lexicographically. For agents  $i_0, \dots, i_\ell$  and goods  $j_1, \dots, j_\ell$ , consider a path  $P = (i_0, j_1, i_1, j_2, \dots, j_\ell, i_\ell)$  in the supplemented MBB graph, where for all  $1 \leq \ell' \leq \ell$ ,  $j_{\ell'} \in mbb_{(i_{\ell'-1})} \cap \mathbf{x}_{\ell'}$ . Define the *level* of an agent  $h$  to be the length of the shortest such path from the LS to  $h$ , and to be  $n$  if no such path exists. Define *alternating paths* to be such paths beginning with agents at a lower level and ending with agents at a strictly higher level. The edges in an alternating path alternate between MBB edges and allocation edges.

---

**Algorithm 1.** EFX+fPO allocation for  $\{a, b\}$ -instances

---

**Input:** Fair division  $\{a, b\}$ -instance  $(N, M, V)$

**Output:** An integral allocation  $\mathbf{x}$

- 1: Scale values to  $\{1, k\}$ , where  $k = a/b > 1$ .
  - 2:  $(\mathbf{x}, \mathbf{p}) \leftarrow$  Integral welfare-maximizing market allocation, where  $p_j = v_{ij}$  for  $j \in \mathbf{x}_i$ .
  - 3: Let  $i \in \operatorname{argmin}_{h \in N} \mathbf{p}(\mathbf{x}_h)$  be the least spender
  - 4: **if** there is an alternating path  $(i, j_1, i_1, \dots, j_\ell, i_\ell)$ , s.t.  $\mathbf{p}(\mathbf{x}_{i_\ell} \setminus \{j_\ell\}) > \mathbf{p}(\mathbf{x}_i)$  **then**
  - 5:     Transfer  $j_\ell$  from  $i_\ell$  to  $i_{\ell-1}$
  - 6:     Repeat from Line 3
  - 7: **if**  $\forall$  agents  $h \notin C_i$ , and  $\forall j \in \mathbf{x}_h : \mathbf{p}(\mathbf{x}_h \setminus \{j\}) \leq \mathbf{p}(\mathbf{x}_i)$  **then return**  $\mathbf{x}$      ▷ pEFX condition satisfied for all agents not in component of LS, defined in Def.1
  - 8: **else**
  - 9:     Raise prices of goods in  $C_i$  by a multiplicative factor of  $k$
  - 10:     Repeat from Line 3
- 

**Definition 1 (Component  $C_i$  of a least spender  $i$ ).** For a least spender  $i$ , define  $C_i^\ell$  to be the set of all goods and agents which lie on alternating paths of length  $\ell$ . Call  $C_i = \bigcup_\ell C_i^\ell$  the component of  $i$ , the set of all goods and agents reachable from the least spender  $i$  through alternating paths.

We now describe Algorithm 1. Let  $k = a/b > 1$ . Let us first scale the valuations to  $\{1, k\}$  since both properties EFX and fPO are scale-invariant. The algorithm starts with a welfare maximizing integral allocation  $(\mathbf{x}, \mathbf{p})$ , where  $p_j = v_{ij}$  if  $j \in \mathbf{x}_i$ . The algorithm then explores if there is an alternating path  $P = (i = i_0, j_1, i_1, \dots, j_\ell, i_\ell = h)$ , where  $i$  is the LS agent, such that  $\mathbf{p}(\mathbf{x}_h \setminus \{j_\ell\}) > \mathbf{p}(\mathbf{x}_i)$ , i.e., an alternating path along which the pEF1 condition is violated for the LS agent. We call any such agent  $h$  who owns some good  $j$  such that the pEF1 condition is not satisfied by the LS with respect to good  $j$ , a pEF1-violator. When such a path is encountered, the algorithm *transfers*  $j_\ell$  from  $h$  to  $i_{\ell-1}$ . This process is repeated from Line 3 to account for a possible change in the LS, until there is no such path in the component  $C_i$  of the LS agent. Suppose there is some agent  $h \notin C_i$  for which the pEFX condition is not satisfied with respect to the LS, then the algorithm *raises the prices* of all goods in the component of the LS agent by a factor of  $k$ , and the algorithm proceeds once again from Line 3.

The proof of Theorem 2 relies on Lemmas 1-6. We first show that we can re-scale prices to  $\{1, k\}$ .

**Lemma 2.** For every outcome  $(\mathbf{x}, \mathbf{p})$  constructed during the run of Algorithm 1, there exists a set of prices  $\mathbf{q}$  such that  $(\mathbf{x}, \mathbf{q})$  is also on MBB, and for every  $j \in M$ ,  $q_j \in \{1, k\}$ .

*Proof.* Note that initially all prices are either 1 or  $k$ . Since all price rises are by a factor of  $k$  (Line 9), final prices are of the form  $p_j = k^{s_j}$ , for  $s_j \in \mathbb{Z}_{\geq 0}$ . Let  $j_0$  be the smallest priced good with  $p_{j_0} = k^s$ , and let  $j_0 \in \mathbf{x}_i$ , for some agent  $i \in N$ . Then  $\forall j \in \mathbf{x}_i : p_j \in \{k^s, k^{s+1}\}$ . By the MBB condition for any agent  $h \neq i$  for



$j' \in \mathbf{x}_h$  and  $j \in \mathbf{x}_i$ :

$$\frac{v_{hj'}}{p_{j'}} \geq \frac{v_{hj}}{p_j},$$

which gives:

$$p_{j'} \leq \frac{v_{hj'}}{v_{hj}} p_j \leq k^{s+2}.$$

Thus all  $p_j \in \{k^s, k^{s+1}, k^{s+2}\}$ . Either all  $p_j \in \{k^s, k^{s+1}\}$ , or  $\exists j \in \mathbf{x}_h$  with  $p_j = k^{s+2}$ , for some agent  $h \in N$ . Then by the MBB condition for any good  $j'$ :

$$\frac{v_{hj}}{p_j} \geq \frac{v_{hj'}}{p_{j'}},$$

which gives:

$$p_{j'} \geq \frac{v_{hj'}}{v_{hj}} p_j \geq k^{s+1}.$$

Thus either all  $p_j \in \{k^s, k^{s+1}\}$  or all  $p_j \in \{k^{s+1}, k^{s+2}\}$ . In either case we can scale the prices to belong to  $\{1, k\}$ .  $\square$

This in fact shows that at any stage of Algorithm 1, the prices of goods are in  $\{k^s, k^{s+1}\}$  for some  $s \in \mathbb{Z}_{\geq 0}$ . This, along with the fact that goods are always transferred along MBB edges, and the prices are raised only by factor of  $k$ , leads us to conclude that the MBB condition is never violated for any agent and the allocation is always on MBB throughout the run of the algorithm. Hence the allocation is fPO.

**Lemma 3.** *The allocation  $\mathbf{x}$  returned by Algorithm 1 is on MBB w.r.t. prices  $\mathbf{p}$  upon termination. Thus,  $\mathbf{x}$  is fPO.*

The full proof of the above Lemma appears in the appendix. We now show: correctness:

**Lemma 4.** *The allocation  $\mathbf{x}$  returned by Algorithm 1, together with the prices  $\mathbf{p}$  on termination is pEFX.*

*Proof.* To see why  $(\mathbf{x}, \mathbf{p})$  is pEFX, first note that by Lemma 2, we can assume the prices are in  $\{1, k\}$ . Suppose  $(\mathbf{x}, \mathbf{p})$  is not pEFX. Then there must be an agent  $h$  and some good  $j \in \mathbf{x}_h$  s.t.  $\mathbf{p}(\mathbf{x}_h \setminus \{j\}) > \mathbf{p}(\mathbf{x}_i)$ , where  $i$  is the least spender. If  $h \notin C_i$ , the algorithm would not have halted (negation of condition in line 8 holds). Therefore  $h$  is in  $C_i$ . Since the algorithm has halted, this means that along all alternating paths  $(i, j_1, i_1, \dots, h', j, h)$ , it is the case that  $\mathbf{p}(\mathbf{x}_h \setminus \{j\}) \leq \mathbf{p}(\mathbf{x}_i)$ . Suppose there is some alternating path s.t.  $p_j = 1$ . We know for all  $j' \in \mathbf{x}_h$ ,  $p_{j'} \geq 1$ . Thus:

$$\mathbf{p}(\mathbf{x}_i) \geq \mathbf{p}(\mathbf{x}_h \setminus \{j\}) = \mathbf{p}(\mathbf{x}_h) - 1 \geq \mathbf{p}(\mathbf{x}_h \setminus \{j'\}),$$

which means that  $i$  is pEFX towards  $h$ . Now suppose along all alternating paths  $(i, j_1, i_1, \dots, h', j, h)$ , it holds that  $p_j = k$ . Since  $(\mathbf{x}, \mathbf{p})$  is not pEFX, it must be the case that there is some good  $j' \in \mathbf{x}_h$  that is not reachable from  $i$  via

any alternating path, with  $p_{j'} = 1$ . This means that  $j' \notin mbb_{h'}$ . Since  $j \in mbb_{h'}$ , comparing the bang-per-buck ratios gives  $v_{h'j}/p_j > v_{h'j'}/p_{j'}$ . This implies  $v_{h'j} > kv_{h'j'}$  which is not possible when  $v_{h'j}, v_{h'j'} \in \{1, k\}$ , thus leading to a contradiction. Hence we conclude that  $(\mathbf{x}, \mathbf{p})$  is pEFX.  $\square$

**Lemma 5.** *Algorithm 1 terminates in polynomial-time.*

*Proof.* (Sketch) We first note that the number of alternating paths from an agent  $i$  to an agent  $h$  who owns a good  $j$  which is then transferred to an agent  $h'$  is at most  $n \cdot n \cdot m$ . Thus there are at most  $\text{poly}(n, m)$  transfers with the same LS and no price rise step.

Next, we argue that the number of identity changes of the least spender without a price rise step is  $\text{poly}(n, m)$ . Suppose an agent  $i$  ceases to be the LS at iteration  $t$ , and subsequently (without price-rise steps) becomes the LS again for the first time at time  $t'$ . We show that the spending of  $i$  is strictly larger at  $t'$  than at  $t$ , and hence has strictly larger utility. Since all utility values are integers, the increase in  $i$ 's utility is by at least 1. In any allocation  $\mathbf{x}$ , if  $s_i$  (resp.  $t_i$ ) is the number of goods in  $\mathbf{x}_i$  that are valued at  $b$  (resp.  $a$ ) by  $i$ , the utility of  $i$  is  $u_i = s_i b + t_i a$ . Since  $0 \leq s_i, t_i \leq m$ , the number of different utility values  $i$  can get in any allocation is at most  $O(m^2)$ . Thus, for any agent  $i$ , the number of times her utility increases is at most  $O(m^2)$ . This is our key insight. It implies that without price rises, any agent can become the least spender only  $O(m^2)$  times. Hence, the number of identity changes of the LS in the absence of price rise steps is at most  $O(nm^2)$ .

For polynomial run-time, it remains to be shown that the number of price-rises is  $\text{poly}(n, m)$ . We do this via a potential function argument similar to [20]. The full proof is present in the appendix.  $\square$

Finally, we show that the allocation returned by our algorithm also provides a good approximation to the MNW, and defer the proof to the appendix.

**Lemma 6.** *Let  $\mathbf{x}$  be the allocation output by Algorithm 1. If  $\mathbf{x}^*$  is an MNW allocation, then  $\text{NW}(\mathbf{x}) \geq \frac{1}{1.061} \text{NW}(\mathbf{x}^*)$ .*

*Proof.* (Sketch) Let  $\mathbf{p}$  be the price vector on termination. Consider a scaled fair division instance  $I' = (N, M, V')$  with identical valuations, where  $v'_{ij} = p_j$  for each  $i \in N, j \in M$ . Since  $(\mathbf{x}, \mathbf{p})$  is pEFX for the instance  $I$  (Lemma 4),  $\mathbf{x}$  is EFX for the instance  $I'$ . Barman et al., [6] showed that for identical valuations, any EFX allocation provides a 1.061-approximation to the maximum Nash welfare. Hence  $\mathbf{x}$  provides a 1.061-approximation to the MNW of  $I'$ , and we can show that because  $(\mathbf{x}, \mathbf{p})$  is on MBB (from Lemma 3),  $\mathbf{x}$  gives the same guarantee for the MNW of the instance  $I$ .  $\square$

Lemmas 1, 3, 4, 5, and 6 together prove Theorem 2.

### 3.2 EFX+PO for 3-Valued Instances

On generalizing the class of valuations slightly to  $\{0, a, b\}$ , EFX+PO allocations are no longer guaranteed to exist [34] (see the appendix for an example).

Therefore we investigate the complexity of checking if an EFX and PO allocation exists or not, and show that this problem is NP-hard.

**Theorem 3.** *Given a fair division instance  $I = (N, M, V)$ , checking if  $I$  admits an allocation that is both EFX and PO is NP-hard, even when  $I$  is a  $\{0, a, b\}$ -instance.*

We reduce from 2P2N3SAT, an instance of which consists of a 3SAT formula over  $n$  variables  $\{x_i\}_{i \in [n]}$  in conjunctive normal form, and  $m$  distinct clauses  $\{C_j\}_{j \in [m]}$ , with three literals per clause. Additionally, each variable  $x_i$  appears exactly twice negated and exactly twice unnegated in the formula. Deciding if there exists a satisfying assignment for such a formula is known to be NP-complete [7]. Given a 2P2N3SAT-instance  $(\{x_i\}_{i \in [n]}, \{C_j\}_{j \in [m]})$ , we construct a fair division instance with  $2n + m$  agents and  $7n + m$  goods, with all values in  $\{0, 1, 3\}$  as follows:

1. For every variable  $x_i$ , create two agents  $T_i$  and  $F_i$ . Also create 7 goods:  $d_i^T, d_i^F, g_i, y_i^T, z_i^T, y_i^F, z_i^F$ . Both  $T_i$  and  $F_i$  value  $g_i$  at 3.  $T_i$  values  $d_i^T, y_i^T, z_i^T$  at 1, and  $F_i$  values  $d_i^F, y_i^F, z_i^F$  at 1.  $T_i$  and  $F_i$  value all other goods at 0.
2. For every clause  $C_j = \ell_1 \vee \ell_2 \vee \ell_3$ , create one agent  $D_j$  and a good  $e_j$ .  $D_j$  values  $e_j$  at 1. If for any  $k \in [3]$ ,  $\ell_k = x_i$  for some  $i \in [n]$  then  $D_j$  values  $y_i^T, z_i^T$  at 1; and if for any  $k \in [3]$ ,  $\ell_k = \neg x_i$  for some  $i \in [n]$  then  $D_j$  values  $y_i^F, z_i^F$  at 1.  $D_j$  values all other goods at 0.

We show that this instance admits an EFX+PO allocation iff the formula has a satisfying assignment. We illustrate the correspondence between PO allocations and assignments, and how our construction enforces EFX allocations to give rise to satisfying assignments (and vice versa). In any PO allocation, for every  $i \in [n]$ ,  $d_i^A$  must be assigned to  $A_i$ , for  $A \in \{T, F\}$ ; and  $g_i$  must be assigned to  $T_i$  or  $F_i$ . Consider the assignment  $x_i = A$ , if  $g_i$  is allotted to  $A_i$ , for  $A \in \{T, F\}$ , for all  $i \in [n]$ . Suppose for some  $i \in [n]$ ,  $g_i$  is allocated to  $T_i$ . Then in an EFX allocation, because  $F_i$  must not envy  $T_i$  after removing  $d_i^T$  from the bundle of  $T_i$ ,  $F_i$  must get utility at least 3. This is only possible if both  $y_i^F$  and  $z_i^F$  are allocated to  $F_i$ . This leaves  $y_i^T, z_i^T$  for the clause agents, when  $x_i$  is True. Thus if there is a satisfying assignment, the remaining goods can be allocated to the clause agents in an EFX+PO manner. Also, if all assignments are unsatisfying, some clause agent will end up not being EFX towards another agent in any PO allocation.

We defer the full proof to the appendix, and also show:

**Lemma 7.** *Given a fair division instance  $I = (N, M, V)$ , checking if  $I$  admits an allocation that is both EFX and fPO is NP-complete, even when  $I$  is a  $\{0, a, b\}$ -instance.*

## 4 Computing EQX+PO Allocations

We now consider relaxations of the fairness notion of equitability, which demands that all agents receive roughly the same utility. An allocation is said to be

---

**Algorithm 2.** EQX+fPO allocation for  $\{a, b\}$ -instances

---

**Input:** Fair division  $\{a, b\}$ -instance  $(N, M, V)$

**Output:** An integral allocation  $\mathbf{x}$

- 1:  $(\mathbf{x}, \mathbf{p}) \leftarrow$  Integral welfare-maximizing market allocation, where  $p_j = v_{ij}$  for  $j \in \mathbf{x}_i$ .
  - 2: Let  $i \in \operatorname{argmin}_{h \in N} v_h(\mathbf{x}_h)$  be the least utility (LU) agent
  - 3: **if** there is an alternating path  $(i, j_1, i_1, \dots, j_\ell, i_\ell)$ , s.t.  $v_{i_\ell}(\mathbf{x}_{i_\ell} \setminus \{j_\ell\}) > v_i(\mathbf{x}_i)$  **then**
  - 4: Transfer  $j_\ell$  from  $i_\ell$  to  $i_{\ell-1}$
  - 5: Repeat from Line 2
  - 6: **if**  $\forall$  agents  $h \notin C_i$ , and  $\forall j \in \mathbf{x}_h : v_h(\mathbf{x}_h \setminus \{j\}) \leq v_i(\mathbf{x}_i)$  **then return**  $\mathbf{x} \triangleright$  EQX condition satisfied for all agents not reachable through alt. paths from LU agent;  $C_i$  is defined in Def. 1
  - 7: **else**
  - 8: Raise prices of goods in  $C_i$  by a multiplicative factor of  $a/b$
  - 9: Repeat from Line 2
- 

equitable up to any good (EQX) if for all  $i, h \in N$ , for all  $j \in \mathbf{x}_h$  we have  $v_i(\mathbf{x}_i) \geq v_h(\mathbf{x}_h \setminus \{j\})$ . It is known that for binary instances, EQX+PO allocations can be computed in polynomial-time, whenever they exist [20]. Hence we first consider  $\{a, b\}$ -instances. We show that:

**Theorem 4.** *Given a fair division  $\{a, b\}$ -instance  $I = (N, M, V)$ , an allocation that is both EQX and fPO exists and can be computed in polynomial-time.*

We prove this by showing that Algorithm 2 terminates in polynomial-time with an EQX+fPO allocation. Since we are interested in EQX as opposed to EFX, we need not construct a pEFX allocation and can instead work directly with the values. Since the techniques used in the analysis of Algorithm 2 are similar to the analysis of Algorithm 1, we defer the full proof to the appendix. We remark here that our techniques also enable us to show that EQ.1+fPO allocations can be computed in polynomial-time for  $\{a, b\}$ -instances of *chores*.

For  $\{0, a, b\}$ -instances, EQX+PO allocations need not exist (example in appendix). Therefore, we study the complexity of checking if an EQX+PO allocation exists or not, and show that this problem too is NP-hard. The full proof is deferred to the appendix.

**Theorem 5.** *Given a fair division instance  $I = (N, M, V)$ , checking if  $I$  admits an allocation that is both EQX and PO is NP-hard, even when  $I$  is a  $\{0, a, b\}$ -instance.*

## 5 Maximizing Nash Welfare

We turn to the problem of maximizing Nash welfare for  $t$ -valued instances when  $t \in \{2, 3\}$ . Recall that for  $\{a, b\}$ -instances, we showed in Lemma 6 that Algorithm 1 approximates the MNW to a 1.061-factor.

Turning to 3-valued instances, our final result shows APX-hardness for the MNW problem with we slightly generalize the class of allowed values to  $\{0, a, b\}$ .

This rules out the existence of a polynomial-time approximation scheme (PTAS) for the MNW problem even  $\{0, a, b\}$ -instances, thus strengthening the result of [1], who showed NP-hardness for the same. The full proof is present in the appendix.

**Theorem 6.** *Given a fair division instance  $I = (N, M, V)$ , it is NP-hard to approximate the MNW to a factor better than 1.00019, even for  $\{0, a, b\}$ -instances.*

We present the reduction and defer the full proof to the appendix. We consider a 2P2N3SAT-instance:  $\{x_i\}_{i \in [n]}, \{C_j\}_{j \in [m]}$ , where  $3m = 4n$ . For each variable  $x_i$ , we create two agents  $T_i, F_i$  and one good  $g_i$  which is valued at 2 by both  $T_i, F_i$ . For each clause  $C_j$ , we create a good  $h_j$  which is valued at 1 by agent  $A_i$  if setting  $x_i = A$  makes  $C_j$  true, for  $A \in \{T, F\}$ . We also create  $2n - m$  dummy goods  $\{d_j\}_{j \in [2n-m]}$  which are valued at 1 by all agents. All other values are 0. We show that if we can approximate the MNW to a factor better than 1.00019, we can decide if there is an assignment with  $\geq \rho_1 m$  clauses, or all assignments satisfy at most  $\leq \rho_2 m$  clauses, for specific constants  $\rho_1, \rho_2$ . The latter problem is known to be NP-complete [7].

## 6 Conclusion

In this paper, we push the boundary between tractable and intractable cases for the problems of fair and efficient allocations. We presented positive algorithmic results for computing EFX+PO, EQX+PO, and 1.061-approximate MNW allocations for 2-valued instances. In contrast, we showed that for 3-valued instances, checking existence of EFX+PO (or EQX+PO) allocations is NP-complete, and computing MNW is APX-hard. Our techniques can be adapted to compute EQ.1+PO allocations for 2-valued instances of *chores*, and an interesting direction for future work is to see if we can compute EF1+PO allocations in the chores setting, even for 2-valued instances. We also leave open the problem of computing an MNW allocation for general 2-valued instances.

## References

1. Amanatidis, G., Birmpas, G., Filos-Ratsikas, A., Hollender, A., Voudouris, A.A.: Maximum Nash welfare and other stories about EFX. In: Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence (IJCAI), pp. 24–30 (2020)
2. Aziz, H.: The Hylland-Zeckhauser rule under bi-valued utilities. CoRR abs/2006.15747 (2020)
3. Aziz, H., Biró, P., Lang, J., Lesca, J., Monnot, J.: Efficient reallocation under additive and responsive preferences. *Theor. Comput. Sci.* **790**, 1–15 (2019)
4. Barman, S., Krishnamurthy, S.: On the proximity of markets with integral equilibria. In: Proceedings of the 33rd AAAI Conference on Artificial Intelligence, pp. 1748–1755 (2019)

5. Barman, S., Krishnamurthy, S.K., Vaish, R.: Finding fair and efficient allocations. In: Proceedings of the 19th ACM Conference on Economics and Computation (EC), pp. 557–574 (2018)
6. Barman, S., Krishnamurthy, S.K., Vaish, R.: Greedy algorithms for maximizing Nash social welfare. In: Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems (AAMAS), pp. 7–13 (2018)
7. Berman, P., Karpinski, M., Scott, A.: Approximation hardness of short symmetric instances of MAX-3SAT. Electronic Colloquium on Computational Complexity (ECCC) (Jan 2003)
8. Bliem, B., Bredereck, R., Niedermeier, R.: Complexity of efficient and envy-free resource allocation: few agents, resources, or utility levels. In: Proceedings of the 25th International Joint Conference on Artificial Intelligence (IJCAI), pp. 102–108 (2016)
9. Bogomolnaia, A., Moulin, H.: Random matching under dichotomous preferences. *Econometrica* **72**(1), 257–279 (2004)
10. Brams, S., Taylor, A.: Fair Division: From Cake-Cutting to Dispute Resolution. Cambridge University Press, Cambridge (1996)
11. Budish, E.: The combinatorial assignment problem: Approximate competitive equilibrium from equal incomes. *J. Polit. Econ.* **119**(6), 1061–1103 (2011)
12. Caragiannis, I., Kurokawa, D., Moulin, H., Procaccia, A.D., Shah, N., Wang, J.: The unreasonable fairness of maximum Nash welfare. In: Proceedings of the 17th ACM Conference on Economics and Computation (EC), pp. 305–322 (2016)
13. Chakrabarty, D., Khanna, S., Li, S.: On (1, e)-restricted assignment makespan minimization. In: Proceedings of the 26th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA), pp. 1087–1101 (2015)
14. Chaudhury, B.R., Cheung, Y.K., Garg, J., Garg, N., Hoefer, M., Mehlhorn, K.: On fair division for indivisible items. In: 38th IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS), pp. 1–17 (2018)
15. Chaudhury, B.R., Garg, J., Mehlhorn, K.: EFX exists for three agents. In: Proceedings of the 21st ACM Conference on Economics and Computation (EC), pp. 1–19 (2020)
16. Chaudhury, B.R., Garg, J., Mehlhorn, K., Mehta, R., Misra, P.: Improving EFX guarantees through rainbow cycle number. CoRR abs/2103.01628 (2021). To appear in ACM EC 2021
17. Cole, R., Gkatzelis, V.: Approximating the Nash social welfare with indivisible items. In: Proceedings of the Forty-Seventh Annual ACM Symposium on Theory of Computing (STOC), pp. 371–380 (2015)
18. Darmann, A., Schauer, J.: Maximizing Nash product social welfare in allocating indivisible goods. *SSRN Electron. J.* **247**(2), 548–559 (2014)
19. Foley, D.: Resource allocation and the public sector. *Yale Econ. Essays* **7**(1), 45–98 (1967)
20. Freeman, R., Sikdar, S., Vaish, R., Xia, L.: Equitable allocations of indivisible goods. In: Proceedings of the 28th International Joint Conference on Artificial Intelligence (IJCAI), pp. 280–286 (2019)
21. Garg, J., Hoefer, M., Mehlhorn, K.: Satiation in Fisher markets and approximation of Nash social welfare. CoRR abs/1707.04428 (2017)
22. Garg, J., Hoefer, M., Mehlhorn, K.: Approximating the Nash social welfare with budget-additive valuations. In: Proceedings of the 29th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA), pp. 2326–2340 (2018)

23. Garg, J., Murhekar, A.: On fair and efficient allocations of indivisible goods. In: Proceedings of the 35th AAAI Conference on Artificial Intelligence (2021)
24. Mehlhorn, K.: Assigning papers to referees. In: Albers, S., Marchetti-Spaccamela, A., Matias, Y., Nikolettseas, S., Thomas, W. (eds.) ICALP 2009. LNCS, vol. 5555, pp. 1–2. Springer, Heidelberg (2009). [https://doi.org/10.1007/978-3-642-02927-1\\_1](https://doi.org/10.1007/978-3-642-02927-1_1)
25. Golovin, D.: Max-min fair allocation of indivisible goods. Technical report, CMU-CS-05-144 (2005)
26. Gourvès, L., Monnot, J., Tlilane, L.: Near fairness in matroids. In: Proceedings of the 21st European Conference on Artificial Intelligence (ECAI), pp. 393–398 (2014)
27. de Keijzer, B., Bouveret, S., Klos, T., Zhang, Y.: On the complexity of efficiency and envy-freeness in fair division of indivisible goods with additive preferences. In: Rossi, F., Tsoukias, A. (eds.) ADT 2009. LNCS (LNAI), vol. 5783, pp. 98–110. Springer, Heidelberg (2009). [https://doi.org/10.1007/978-3-642-04428-1\\_9](https://doi.org/10.1007/978-3-642-04428-1_9)
28. Lee, E.: APX-hardness of maximizing Nash social welfare with indivisible items. *Inf. Process. Lett.* **122**, 17–20 (07 2015)
29. Lipton, R.J., Markakis, E., Mossel, E., Saberi, A.: On approximately fair allocations of indivisible goods. In: Proceedings of the 5th ACM Conference on Electronic Commerce (EC), pp. 125–131 (2004)
30. Mas-Colell, A., et al.: *Microeconomic Theory*. Oxford University Press, Oxford (1995)
31. McGlaughlin, P., Garg, J.: Improving Nash social welfare approximations. *J. Artif. Intell. Res.* **68**, 225–245 (2020)
32. Moulin, H.: *Fair Division and Collective Welfare*. MIT Press, Cambridge (2004)
33. Murhekar, A., Garg, J.: On fair and efficient allocations of indivisible goods. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 35, no. 6, pp. 5595–5602 (2021)
34. Plaut, B., Roughgarden, T.: Almost envy-freeness with general valuations. In: Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA), pp. 2584–2603 (2018)
35. Procaccia, A.D.: Technical perspective: an answer to fair division’s most enigmatic question. *Commun. ACM* **63**(4), 118 (2020)
36. Steinhaus, H.: Sur la division pragmatique. *Econometrica* **17**(1), 315–319 (1949)
37. Varian, H.R.: Equity, envy, and efficiency. *J. Econ. Theory* **9**(1), 63–91 (1974)
38. Vazirani, V.V., Yannakakis, M.: Computational complexity of the Hylland-Zeckhauser scheme for one-sided matching markets. In: Proceedings of the 12th Innovations in Theoretical Computer Science Conference (ITCS) (2021)
39. Woeginger, G.J.: A polynomial-time approximation scheme for maximizing the minimum machine completion time. *Oper. Res. Lett.* **20**(4), 149–154 (1997)