

# EFX and PO Allocation Exists for Two Types of Goods

Vladimir Davidiuk<sup>2</sup>, Yuriy Dementiev<sup>1</sup>, Artur Ignatiev<sup>1</sup>, Danil Sagunov<sup>1</sup>

<sup>1</sup>ITMO University

<sup>2</sup>St. Petersburg State University

## Abstract

We study the problem of fairly and efficiently allocating indivisible goods among agents with additive valuations. We focus on envy-freeness up to any good (EFX) — an important fairness notion in fair division of indivisible goods. A central open question in this field is whether EFX allocations always exist for any number of agents. While recent results have established EFX existence for settings with at most three distinct valuations and for two types of goods, the general case remains unresolved.

In this paper, we extend the existent knowledge by proving that EFX allocations satisfying Pareto optimality (PO) always exist and can be computed in quasilinear time when there are two types of goods, given that the valuations are positive. Our findings demonstrate a fairly simple and efficient algorithm constructing an EFX+PO allocation.

## Introduction

Fair division of indivisible goods is a core research area in algorithmic game theory and computational social choice, focusing on the equitable allocation of discrete items—such as property, licenses, or humanitarian aid—among agents with heterogeneous preferences. The online platform Spliddit ([spliddit.org](https://spliddit.org)) offers practical implementations of fair division algorithms, addressing real-world allocation challenges including rent distribution among roommates, taxi fare splitting, and fair assignment of goods between individuals. A central goal is to achieve fairness and effectiveness guarantees that balance efficiency and equity, even when exact solutions are theoretically or computationally out of reach.

The gold standard of fairness, envy-freeness (EF) (Foley 1966), ensures no agent prefers another’s allocation over their own. Yet, EF allocations often fail to exist for indivisible goods, and even when they do, identifying them is NP-complete as shown by (Lipton et al. 2004) even with two agents. This limitation has spurred the study of meaningful relaxations. The initial relaxation of envy-freeness is the concept of envy-freeness up to one good (EF1), which was informally introduced by (Lipton et al. 2004) and later formally defined by (Budish 2011). Under EF1, an agent  $i$  may envy agent  $j$ , provided there is at least one good in  $j$ ’s

bundle such that, if removed,  $i$ ’s envy toward  $j$  would disappear. EF1 allocations always exist for indivisible goods under additive valuations and can be computed in polynomial time using algorithms like the envy cycle elimination method (Lipton et al. 2004). (Caragiannis et al. 2019) proved that Round-Robin algorithms leads to an EF1 allocation. As proven by (Caragiannis et al. 2019), allocations maximizing the Nash Social Welfare (MNW) are necessarily Pareto-efficient and envy-free up to one good. Among these, envy-freeness up to any good (EFX), introduced by (Caragiannis et al. 2019), has emerged as a compelling alternative. EFX softens the strictness of EF: while an agent may envy another’s bundle, this envy vanishes upon the removal of any single good from the envied bundle. The existence of EFX is still open for  $n$  agents with arbitrary additive valuations.

Recent years have seen significant progress in answering this question, though a general solution remains elusive. The first positive existence result was shown by (Plaut and Roughgarden 2020) for when the agents have identical valuations, they showed that the lexicographic minimal (lexmin) solution guarantees EFX in this case. For two agents, (Plaut and Roughgarden 2020) showed that the cut and choose algorithm guarantees EFX. (Chaudhury, Garg, and Mehlhorn 2024) extended this result to three agents, while (Mahara 2023) and (Prakash HV et al. 2025) proved existence for settings with two and three types of agents, respectively. On the other side, (Gorantla, Marwaha, and Velusamy 2023) demonstrated that EFX allocations exist when goods belong to just two distinct types, even providing a partial characterization of when exact envy-freeness (EF) is achievable in such cases. A previous paper established the existence of EFX and PO allocations under lexicographic preferences (Hosseini et al. 2021). Moreover, (Garg and Murhekar 2023) proved that an EFX and fPO allocation exists and can be computed in polynomial time for bivalued instances.

Pareto optimality is a fundamental efficiency concept in fair division, ensuring no agent can improve their allocation without harming others. (Amanatidis et al. 2021) showed that when goods take on at most two possible values, the Maximum Nash Welfare (MNW) solution guarantees EFX and Pareto optimality (PO), also they show a polynomial-time algorithm for EFX based on matching. However, (Plaut and Roughgarden 2020) also revealed a sobering limitation: if agents can assign zero value to goods, EFX and PO may

be incompatible. This leaves an open problem: Does an EFX and PO allocation always exist when all valuations are positive?

**Our Contribution.** We significantly advance the understanding of fair and efficient allocations for two types of goods. First, we prove that EFX+PO allocation always exist when all agents have positive utilities, while demonstrating that the stronger EFX+fPO combination remains impossible. This improves upon (Gorantla, Marwaha, and Velusamy 2023), who established EFX existence without Pareto optimality.

We provide an efficient algorithmic solution: our approach computes such an allocation in  $\mathcal{O}(n \log n + \log m)$  time for  $n$  agents and  $m$  goods. It works as fast as  $\mathcal{O}(\log n + \log m)$  time, if the agents are ordered by their relative valuations of good types. Additionally, we introduce a class of *proper* allocations and prove that every *proper* allocation is Pareto optimal, offering new structural insights into the interplay between fairness and efficiency.

**Example: EFX and fPO are not always compatible.**

Consider a simple example with two types of goods (croissant and coffee) and two agents (Alice (1) and Bob (2)). There are two croissants ( $a$ ) and two coffees ( $b$ ), with valuations  $v_1(a) = v_2(a) = 1$  and  $v_1(b) = 10$  and  $v_2(b) = 9$ . In the unique EFX allocation, each agent receives one croissant and one coffee. While this allocation is envy-free up to any good, it fails to be fractionally Pareto optimal (fPO): a fractional allocation where Alice gets 1.1 coffee and Bob gets 0.9 coffee plus two croissant would strictly improve the allocation, Alice has the same utility, but Bob increases his utility (11 for Alice and 10.1 for Bob). However, in the indivisible setting, this EFX allocation is Pareto optimal (PO), as no discrete reallocation can improve one agent’s utility without harming the other. This demonstrates that there may not be allocations that satisfy EFX and fPO simultaneously. In our turn, we show that an EFX+PO allocation always exists.







				
	1	1	10	10
	1	1	9	9

Table 1: Example with two types of goods.

**Additional Related Work.** The fair division of indivisible goods has witnessed remarkable theoretical and algorithmic progress in recent years, as documented in several comprehensive surveys (Amanatidis et al. 2023; Aziz et al. 2022; Nguyen and Rothe 2023). One promising direction involves EFX relaxation by allowing a small subset of goods to remain unassigned—often interpreted as charitable donations—while maintaining fairness guarantees. While trivially satisfying envy-freeness by leaving all goods unallocated is meaningless, meaningful progress has

been made in bounding both the number of discarded items and their welfare impact. (Caragiannis, Gravin, and Huang 2019) demonstrated that an EFX allocation exists for a subset of goods while preserving at least half of the Maximum Nash Welfare. (Chaudhury et al. 2021) developed an algorithm for computing partial EFX allocations where: (i) at most  $n - 1$  goods remain unassigned, and (ii) no agent strictly prefers the set of unallocated goods to their own bundle, while (Berger et al. 2022) and (Mahara 2024) later reduced this to  $n - 2$  goods in general and just one good for four agent instances. (Ghosal et al. 2025) generalized these results, proving that for agents with at most  $k$  distinct valuations, EFX allocations exist with  $k - 2$  unassigned goods.

Approximate EFX notions have also gained attention, where an allocation is  $\alpha$ -EFX if no agent envies another after scaling the other’s bundle without any good by  $\alpha$ , formally if for every pair of agents  $i$  and  $j$ :  $v_i(X_i) \geq \alpha \cdot v_i(X_j \setminus g)$  for any  $g \in X_j$ . (Plaut and Roughgarden 2020) established the existence of 0.5-EFX allocations, and this bound was later improved to 0.618 by (Amanatidis, Markakis, and Ntokos 2020). Most recently, (Amanatidis, Filos-Ratsikas, and Sgouritsa 2024) showed that  $\frac{2}{3}$ -EFX allocations exist for up to seven agents or when agents have no more than three distinct valuations.

**Setting.** The setting consists of a set of  $n$  agents  $N = [n] = \{1, \dots, n\}$ , and a multiset  $M$  of indivisible goods that contains copies of two different goods:  $g_1$  with multiplicity  $m_1$  and  $g_2$  with multiplicity  $m_2$ . Each agent  $i \in N$  has a valuation function  $v_i: 2^M \rightarrow \mathbb{R}_{>0}$ , which quantifies the utility that  $i$  derives from any subset of goods. We focus on additive valuations, meaning the value of a subset of goods is simply the sum of the values of its individual items. Formally, for any subset  $S \subseteq M$ , the valuation function satisfies  $v_i(S) = \sum_{g \in S} v_i(g)$ . An allocation  $X = (X_1, X_2, \dots, X_n)$  is a partition of  $M$  into  $n$  disjoint subsets, called bundles, where each agent  $i$  receives the bundle  $X_i$ . We match allocation  $X$  with the multiplicity of goods in each agent’s bundle  $\{(x_{i,1}, x_{i,2}) \mid i \in [n]\}$ , so we can assume that agents utility equals  $v_i(X_i) = x_{i,1} \cdot v_{i,1} + x_{i,2} \cdot v_{i,2}$ , where  $v_{i,j} = v_i(\{g_j\})$  for each  $i \in [n], j \in [2]$ .

**Definition 1 (EFX).** An allocation  $X$  is envy-free up to any good (EFX) if, for every pair of agents  $i, j \in N$ , it holds that  $v_i(X_i) + v_i(g) \geq v_i(X_j)$  for any  $g \in X_j$ .

We say that agent  $i$  envies agent  $j$  up any good if  $v_i(X_i) + \min_{g \in X_j} v_i(g) < v_i(X_j)$ .

**Definition 2 (PO).** An allocation  $X$  is Pareto optimal (PO) if there is no allocation  $Y$  such that  $v_i(Y_i) \geq v_i(X_i)$  for all  $i \in N$  and  $v_j(Y_j) > v_j(X_j)$  for some  $j \in N$ . Equivalently, we will say that such an allocation is not Pareto dominated by any other allocation.

## The Setup

To show our main result, we have to define several crucial concepts. In this section, we give necessary definitions and formulate fundamental properties of our allocation structures. They serve as a backbone of our algorithm.

Most of the properties in this section are given without proofs to ease the understanding of the construction and

the algorithm itself. Complete proofs of such properties and claims can be found in subsequent sections or the supplementary materials of this paper. We mark all statements, the proof of which can be found in the appendix, with a star ( $\star$ ).

**Input preprocessing.** Our first goal is to make the input data much more convenient and reflecting the actual allocation properties.

By  $n$  we denote the number of agents, by  $m_1$  and  $m_2$  we denote the total quantities of items of first and second types respectively. We naturally assume that  $n \geq 1$  and  $m_1, m_2 \geq 1$ . Agents are identified with integer numbers in  $[n]$ . For  $i \in [n]$ ,  $j \in [2]$ , let  $v_{i,j}$  denote the utility of an item of type  $j$  as estimated by agent  $i$ .

We then divide the agents in two groups, the first group consists of agents  $i$  that have  $v_{i,2} \leq v_{i,1}$  (agents that prefer a good of type 1 to a good of type 2 or value them equally), the second consists of agents  $i$  that have  $v_{i,2} > v_{i,1}$ . Note that there may be several ways to divide the agents, choose any of them arbitrarily. Let  $n_1, n_2$  denote the obtained group sizes. If it appears that  $\frac{m_1}{n_1} < \frac{m_2}{n_2}$ , we interchange the good types and the groups (that is why we defined the groups symmetrically). Then the inequality  $\frac{m_1}{n_1} \geq \frac{m_2}{n_2}$  holds. If  $n_2 = 0$ , we consider that  $\frac{m_2}{n_2} = \infty$ . This means, in particular, that  $n_2 > 0$ .

We assume that  $v_{i,j} > 0$  for each  $i \in [n], j \in [2]$ , as motivated in the introduction. Then, we normalize the agent utilities: we replace  $(v_{i,1}, v_{i,2})$  with  $(1, v_{i,2}/v_{i,1})$  for each agent  $i \in [n]$ . Finally, we re-enumerate the agents in the way that  $v_{i,2} \leq v_{i+1,2}$  holds for each  $i \in [n-1]$ . By definition of  $n_1$  and  $n_2$ , we have that  $\forall i \in [n_1] : v_{i,2} \leq 1$  and  $v_{n_1+1,2} \geq 1$ .

The other thing crucial to our further constructions is that the total number of goods is at least the number of agents.

**Observation 1 ( $\star$ ).** *If  $m_1 + m_2 \leq n$ , give one good of the first type to each agent  $i$  with  $i \leq m_1$ . Give one good of the second type to each agent  $i$  with  $i \geq n - m_2 + 1$ . This allocation is EFX+PO.*

We are ready to summarize our restrictions on the input, that all hold true after the input is preprocessed. We provide them below.

### Preprocessed input restrictions

- For each  $i \in [n]$ ,  $v_{i,1} = 1$  and  $v_{i,2} > 0$ ;
- For each  $i \in [n-1]$ ,  $v_{i,2} \leq v_{i+1,2}$ ;
- $n_1 + n_2 = n$  and  $v_{n_1,2} \leq 1$  and  $v_{n_1+1,2} \geq 1$ ;
- $m_1/n_1 \geq m_2/n_2$  and  $m_2 > 0$  and  $n_2 > 0$ ;
- $m_1 + m_2 \geq n$ .

Throughout the rest of the paper, we always assume that the input satisfies the above constraints. Our results might not hold true if these constraints are not satisfied.

**Proper allocations.** In our work, the Pareto-optimality is achieved in all of the constructed allocations, both intermediate ones (that might not be EFX) and the resulting one (that is necessary EFX). All of our allocations fall under the notion given below.

<sup>1</sup>We do not use  $v_{i,2} > v_{i,1}$  here for purpose.

**Definition 3 (Proper allocation).** An allocation of goods  $\{(x_{i,1}, x_{i,2})\}$  is *proper*, if there exists  $t \in [n]$  such that

- for each  $i \in [t-1]$ ,  $x_{i,2} = 0$ , and
- for each  $i \in [t+1, n]$ ,  $x_{i,1} < v_{t,2}$ .

For example, the allocation in Observation 1 is proper (choose  $t = m_1$ ). As announced above, we will prove that all proper allocations are PO, as formulated below.

**Theorem 1.** *If allocation is proper, then it is Pareto-optimal.*

**Prioritized equitable allocations.** The following notion is basic to our allocation constructions. It formally defines the process of dividing the identical goods between specific agents in (almost) equal parts, where some specific agents should receive the greater part.

The notion below defines this process formally. Note that we use it for *partial* allocation of goods, while *complete* allocations are formed via performing several partial allocations consecutively.

**Definition 4 (Prioritized equitable allocation, PEA).** For integers  $c \in [2]$ ,  $q \in [m_t]$ ,  $s \in [n]$ , and a sequence of distinct agents  $a_1, a_2, \dots, a_s \in [n]$ , we say that we give  $q$  goods of type  $c$  *equitably prioritized* to the agents  $a_1, a_2, \dots, a_s$ , if

- for each  $i \in [s]$ , agent  $a_i$  receives either  $\lfloor q/s \rfloor$  or  $\lceil q/s \rceil$  goods of type  $c$ ;
- for each  $1 \leq i < j \leq s$ , agent  $a_j$  receives at least as many goods of type  $c$  as  $a_i$ .

Equivalently, if we give  $q$  goods of type  $c$  to  $s$  agents  $a_1, a_2, \dots, a_s$  equitably prioritized, and  $r$  is the remainder of division of  $q$  by  $s$ , then agents  $a_1, a_2, \dots, a_{s-r}$  receive  $\lfloor q/s \rfloor$  goods each, and agents  $a_{s-r+1}, \dots, a_s$  receive  $\lceil q/s \rceil$  goods each.

**Split allocations.** We move on to the central allocation construction of our algorithm, the *split allocations*. Simply speaking, in a split allocation, only a specified agent  $t$  (the “split point”) can receive goods of both types simultaneously. Agents  $i < t$  or agents  $i > t$  receive only goods of the first type or the second type respectively.

A split allocation is uniquely identified by two integers:  $t \in [n]$  and  $k \in [m_2]$  — the number of goods of the second type given to agent  $t$ . These integers should fit under specific constraints. We define all valid pairs of integers  $\mathcal{T} \subset [n] \times [m_2]$ :

$$\mathcal{T} = \left\{ (t, k) : v_{t,2} \geq 1, t < n, k \leq \frac{m_2}{n-t+1} \right\} \cup \{(n, m_2)\}.$$

Note that  $\mathcal{T}$  is not empty since  $n_2 > 0$  and  $m_2 > 0$ . We now give formal definition to split allocations.

**Definition 5 (Split allocations).** For  $(t, k) \in \mathcal{T}$ , the  $(t, k)$ -*split-allocation* is defined by

1. Give  $k$  goods of the second type to agent  $t$ .
2. Give  $(m_2 - k)$  remaining goods of the second type equitably prioritized to agents  $t+1, t+2, \dots, n$ .
3. Let  $p = \lceil k \cdot v_{t,2} \rceil$ . Give  $\min\{p(t-1), m_1\}$  goods of the first type equitably prioritized to agents  $1, 2, \dots, t-1$ .
4. Give remaining  $\max\{0, m_1 - p(t-1)\}$  goods of the first type equitably prioritized to agents  $1, 2, \dots, t$ .

$(t, k)$ -split-allocation as PEA sum		
Type ( $c$ )	Quantity ( $q$ )	Agent seq. ( $a_1, \dots, a_s$ )
2	$k$	$t$
2	$m_2 - k$	$t + 1, \dots, n$
1	$\min\{m_1, \lceil kv_{t,2} \rceil \cdot (t - 1)\}$	$1, \dots, t - 1$
1	$\max\{m_1 - \lceil kv_{t,2} \rceil \cdot (t - 1)\}$	$1, \dots, t$

Table 2: Expressing  $(t, k)$ -split-allocation as a series of prioritized equitable allocations.

We summarize that the  $(t, k)$ -split-allocation is expressed as a sequence of four prioritized equitable allocations in Table 2. Note that the agent  $t$  can receive 0 goods of the first type, as demonstrated in Table 3.

	1	...	$t - 1$	$t$	$t + 1$	...	$n$
1	$\lfloor \frac{m_1}{t-1} \rfloor$	...	$\lfloor \frac{m_1}{t-1} \rfloor$	0	0	...	0
2	0	...	0	$k$	$\lfloor \frac{m_2-k}{n-t} \rfloor$	...	$\lfloor \frac{m_2-k}{n-t} \rfloor$

Table 3: Structure of  $(t, k)$ -split-allocations with  $m_1 \leq p(t - 1)$ . The number in  $i^{\text{th}}$  column and  $j^{\text{th}}$  row equals  $x_{i,j}$ , the number of goods of type  $j$  given to agent  $i$ .

The following simple observation explains why split allocations are PO.

**Observation 2.** For each  $(t, k) \in \mathcal{T}$ , the  $(t, k)$ -split-allocation is proper.

**Envy direction.** Our algorithm will encounter several split allocations. If at least one of split allocations it encounters is EFX, the algorithm will be fine.

But if the  $(t, k)$ -split-allocation is not EFX for  $(t, k) \in \mathcal{T}$ , what can we say about the envy? We will show that our design of split allocations is such that it is not possible that agent  $i < t$  is envious and agent  $j > t$  is envious simultaneously in the  $(t, k)$ -split-allocation. We proceed to formal notions.

**Definition 6** (Left-envious (LE) and right-envious (RE) allocations). We say that an allocation  $X$  is *left-envious*, or *LE* (*right-envious*, or *RE*), if  $X$  is not EFX, and there exists  $i, j$  such that agent  $j$  envies (up to any item) agent  $i$  for  $i < j$  ( $i > j$ ).

We will prove that no  $(t, k)$ -split-allocation can have envy in both directions.

**Theorem 2.** For each  $(t, k) \in \mathcal{T}$ , the  $(t, k)$ -split-allocation cannot be LE and RE simultaneously.

There are two natural “extremal” cases of split allocations. One case is the  $(n, m_2)$ -split-allocation, where all goods of the second type are given to agent  $n$ . This allocation maximizes the total utility provided by goods of the second type. The other case is when goods of the second type receive the leftmost positions possible for a split allocation. This case is the  $(t, \lfloor m_2/(n - t + 1) \rfloor)$ -split-allocation, where  $t \in [n]$  minimum possible such that  $v_{t,2} \geq 1$  and  $m_2 \geq n - t + 1$  (there can be no  $(t', k') \in \mathcal{T}$  with  $t' < t$ ). Following this logic, we define orderings of split allocations.

$(t, k)$ -reallocation as PEA sum		
Type ( $c$ )	Quantity ( $q$ )	Agent seq. ( $a_1, \dots, a_s$ )
2	$k$	$t$
2	$m_2 - k$	$t + 1, \dots, n$
1	$\lceil dv_{t,2} \rceil \cdot (t - 1)$	$1, \dots, t - 1$
1	$\lceil dv_{t,2} \rceil - p$	$t$
1	$m_1 - \lceil dv_{t,2} \rceil \cdot t + p$	$1, \dots, t - 1, t + 1, \dots, \ell, t$

Table 4: Expressing  $(t, k)$ -reallocation as a series of prioritized equitable allocations.

**Definition 7** (Complete ordering of split allocations). A complete linear ordering  $\prec$  on the set  $\mathcal{T}$  is defined by

$$(t_1, k_1) \prec (t_2, k_2)$$

if and only if  $t_1 < t_2$  or  $t_1 = t_2$  and  $k_1 > k_2$ .

The two split allocations described above are naturally maximal and minimal elements in  $\mathcal{T}$  with respect to  $\prec$ . Our second result on envy directions uncovers that these two extremal allocations have specific envy directions (if they are not EFX).

**Theorem 3** (\*). Let  $(t_L, k_L)$  be the minimum element in  $\mathcal{T}$  with respect to  $\prec$ . Let  $(t_R, k_R)$  be the maximum element in  $\mathcal{T}$  with respect to  $\prec$ . Both of the following is true:

1. The  $(t_L, k_L)$ -split-allocation is either EFX or LE,
2. The  $(t_R, k_R)$ -split-allocation is either EFX or RE.

**Reallocation.** To get intuition behind the final concept, assume that none of the split allocations is EFX. Then Theorem 3 guarantees that the  $(t_L, k_L)$ -split-allocation and the  $(t_R, k_R)$ -split-allocation are left-envious and right-envious respectively. Since all split allocations are either LE or RE, there should be  $(t, k) \in \mathcal{T}$  such that the  $(t, k)$ -split-allocation is LE, while for its immediate successor (w.r.t.  $\prec$ )  $(t', k') \in \mathcal{T}$ , the  $(t', k')$ -split-allocation is RE. That is,  $(t, k)$  and  $(t', k')$  form a point where the “envy direction changes”.

This can be, for example,  $(t, k) \in \mathcal{T}$  for  $t < n$  and  $k \geq 2$ , and  $(t, k - 1) \in \mathcal{T}$ , such that the  $(t, k)$ -split-allocation is LE and the  $(t, k - 1)$ -split-allocation is RE.

Our final allocation construction slightly transforms such  $(t, k)$ -split-allocations by redistributing some goods of the first type from agents to the left of  $t$  to agents to the right of  $t$ . We now define this construction formally (see also Table 4).

**Definition 8** ( $(t, k)$ -reallocation). For  $(t, k) \in \mathcal{T}$  such that  $t < n$  and the  $(t, k)$ -split-allocation  $X$  satisfies

$$m_1 \geq \lceil d \cdot v_{t,2} \rceil \cdot t - p,$$

where  $p = \lceil kv_{t,2} \rceil$  and  $d = x_{t+1,2} = \lfloor (m_2 - k)/(n - t) \rfloor$ , the  $(t, k)$ -reallocation is obtained by

1. Start from  $X$ , but take away all items of the first type from each agent.
2. Let  $\ell \in [n]$  be maximum possible such that  $x_{\ell,2} = d$ .
3. Give  $\lceil d \cdot v_{t,2} \rceil$  items of the first type to each agent in  $[t - 1]$ .

4. Give  $\lceil d \cdot v_{t,2} \rceil - p$  first-type items to agent  $t$ .
5. Give the remaining  $m_1 - \lceil d \cdot v_{t,2} \rceil + p$  items of the first type equitably prioritized to:
  - Agents  $1, 2, \dots, \ell$ , in case  $\lceil dv_{t,2} \rceil - p > (d - k) \cdot v_{t,2}$ ;
  - Agents  $1, 2, \dots, t - 1, t + 1, \dots, \ell, t$ , otherwise.

Note that the definition of  $(t, k)$ -reallocations forces additional constraints on  $(t, k)$ . We will prove the following lemma, that guarantees that  $(t, k)$  satisfies these (and other) constraints, if it corresponds to the point of “envy direction change”.

**Lemma 1** ( $\star$ ). *Let  $(t, k), (t', k') \in \mathcal{T}$  be such that the  $(t, k)$ -split-allocation is left-envious, but  $(t', k')$ -split-allocation is right-envious. Moreover, there is no  $(t'', k'') \in \mathcal{T}$  that satisfies  $(t, k) \prec (t'', k'') \prec (t', k')$ . Then  $t < n$  and*

$$\lceil x_{t+1,2} \cdot v_{t,2} \rceil \cdot t - \lceil kv_{t,2} \rceil + 1 \leq m_1 \leq \lceil x_{t+1,2} \cdot v_{t,2} \rceil \cdot t - \lceil (k-1)v_{t,2} \rceil.$$

The final of our results is the following theorem. If combined with Lemma 1, it grants an EFX+PO allocation.

**Theorem 4** ( $\star$ ). *Let  $X$  be the  $(t, k)$ -reallocation for some  $(t, k) \in \mathcal{T}$ . If*

$$m_1 \leq \lceil x_{t+1,2} \cdot v_{t,2} \rceil \cdot t - \lceil (k-1) \cdot v_{t,2} \rceil,$$

*then  $X$  is EFX+PO.*

As discussed before in this section, the Pareto-optimality will follow from the fact that any  $(t, k)$ -reallocation is proper. We will prove this fact as a part of the proof of Theorem 4.

## The Algorithm

We summarize the constructions and properties of the previous section into an algorithm that constructs the desired allocation. It proves, in particular, that EFX+PO allocations always exist for two types of goods, given that all agents’ utilities are positive.

**Theorem 5.** *Given preprocessed input, an EFX+PO allocation can be found in  $\mathcal{O}(\log n + \log m)$  time.*

*Proof.* We present an algorithm that is based on envy directions of split allocations. Its pseudocode can be found in Algorithm 1, and we will refer to its lines throughout the proof.

We have to clarify first why the algorithm is able to work in logarithmic time, given that even just outputting an allocation ( $2n$  integers) takes linear time. In the following claim, we explain why this is possible; moreover, we can even determine whether a split allocation is EFX in constant time.

**Claim 1.** *Given  $(t, k) \in \mathcal{T}$ , we can determine in  $\mathcal{O}(1)$  time whether the  $(t, k)$ -split-allocation is EFX, left-envious or right-envious.*

*Proof of Claim 1.* By definition of split allocations, there are  $b \leq 5$  distinct bundles that agents receive in the  $(t, k)$ -split-allocation  $X$ . Additionally, there are  $b$  contiguous segments of agent numbers  $[a_1, a_2 - 1], [a_2, a_3 - 1], \dots, [a_b, n]$ , where  $a_1 = 1$  and  $a_1 < a_2 < \dots < a_b$ . Agents within the

---

Algorithm 1: The algorithm constructing EFX+PO allocation on preprocessed inputs.

---

```

1  $t_L \leftarrow$  smallest integer in  $[n]$  with  $(t_L, *) \in \mathcal{T}$ ;
2  $X_L \leftarrow$  the  $(t_L, \lfloor \frac{m_2}{n-t_L+1} \rfloor)$ -split-allocation;
3 if  $X_L$  is EFX then return  $X_L$ ;
4  $t_R \leftarrow n$ ;
5  $X_R \leftarrow$  the  $(t_R, m_2)$ -split-allocation;
6 if  $X_R$  is EFX then return  $X_R$ ;
  /* binary search to find  $t$  */
7 while  $t_R - t_L > 1$  do
8    $t_M \leftarrow \lfloor (t_L + t_R)/2 \rfloor$ ;
9    $X_M \leftarrow$  the  $(t_M, \lfloor \frac{m_2}{n-t_M+1} \rfloor)$ -split-allocation;
10  if  $X_M$  is EFX then return  $X_M$ ;
11  if  $X_M$  is LE then  $t_L \leftarrow t_M$  else  $t_R \leftarrow t_M$ ;
12  $t \leftarrow t_L$  and  $k_L \leftarrow \lfloor \frac{m_2}{n-t+1} \rfloor$  and  $k_R \leftarrow 1$ ;
13 if  $(t, k_R)$ -split-allocation is RE then
  /* binary search to find  $k$  */
14  while  $k_L - k_R > 1$  do
15     $k_M \leftarrow \lfloor (k_L + k_R)/2 \rfloor$ ;
16     $X_M \leftarrow$  the  $(t, k_M)$ -split-allocation;
17    if  $X_M$  is EFX then return  $X_M$ ;
18    if  $X_M$  is LE then  $k_L \leftarrow k_M$  else  $k_R \leftarrow k_M$ ;
19   $k \leftarrow k_L$ ;
20 else
21   $X_R \leftarrow$  the  $(t, k_R)$ -split-allocation;
22  if  $X_R$  is EFX then return  $X_R$ ;
  /* LE changes to RE between  $(t, k_R)$ 
   and  $(t+1, \lfloor m_2/(n-t) \rfloor)$  */
23   $k \leftarrow k_R$ ;
24 return the  $(t, k)$ -reallocation.

```

---

same segment receive exactly the same bundle. Note that  $b$  and  $a_1, a_2, \dots, a_b$  are easily computable in  $\mathcal{O}(1)$  time from  $(t, k)$  following Definition 5, as well as the bundles themselves.

Moreover, if there is envy (up to any item) from agent  $i$  to agent  $j$ , and agents  $i - 1, i, i + 1$  receive exactly the same bundle, then there is necessary envy (up to any item) from agent  $i - 1$  to agent  $j$  or from agent  $i + 1$  to agent  $j$ .

To see the last paragraph, assume agent  $i$  envies agent  $j$  up to any item. Equivalently, there is a choice of integers  $z_1, z_2 \in \{0, 1\}$  such that  $z_1 + z_2 = 1$  and  $z_c \leq x_{j,c}$ , and

$$x_{i,1} + x_{i,2} \cdot v_{i,2} < (x_{j,1} - z_1) + (x_{j,2} - z_2) \cdot v_{i,2},$$

or, equivalently,

$$x_{i,1} - x_{j,1} + z_1 < (x_{j,2} - x_{i,2} - z_2) \cdot v_{i,2}. \quad (1)$$

Since  $v_{i-1,2} \leq v_{i,2} \leq v_{i+1,2}$  and  $x_{i-1,c} = x_{i,c} = x_{i+1,c}$  for each  $c \in [2]$ , we have that (1) holds if we replace  $i$  with either  $i + 1$  (if the right part is non-negative), or  $i - 1$  (if the right part is negative). Equivalently, either agent  $i + 1$  or agent  $i - 1$  envies agent  $j$  up to any item.

It follows that there is no need to check whether an agent  $i$  with  $a_s < i < a_{s+1} - 1$  envies (up to any item) any other agent. The only agents we have to check are the agents with

numbers in  $\{a_1, a_2 - 1, a_2, \dots, a_{b-1}, a_b, n\}$ . There are  $b$  distinct bundles in  $X$ , so for each agent we have to make  $b - 1$  comparisons (whether an agent  $i$  envies an agent with this specific bundle).

In total, one should make only a total of  $2b(b-1)$  comparisons between agents and bundles. Each comparison is done in constant time.  $\lrcorner$

We move on to the description of the algorithm itself. The algorithm first evaluates  $t_L$  from the statement of Theorem 3 (Line 1). Since

$$t_L = \max\{\min\{t \in [n] : v_{t,2} \geq 1\}, n - m_2 + 1\},$$

$t_L$  is evaluated in  $\mathcal{O}(\log n)$  time via lower bound binary search over  $(v_{1,2}, v_{2,2}, \dots, v_{n,2})$ .

The other integers from the statement of Theorem 3 are computable in  $\mathcal{O}(1)$  time, and the algorithm checks whether the  $(t_L, k_L)$ - or the  $(t_R, k_R)$ -split-allocation is EFX (Lines 2-5). If the algorithm does not return here, from Theorem 3 we know that the allocations are LE and RE respectively.

The algorithm then aims to find  $(t, k) \in \mathcal{T}$  satisfying Lemma 1. This is done via two binary searches. In the first binary search, the algorithm finds  $t$ , and in the second one — the algorithm finds  $k$  (given that  $t$  is known). During these searches, the algorithm might encounter a split allocation that is EFX. If this happens, the algorithm returns such allocation as its final answer.

We move on to discussion of the first binary search. To perform this search, the algorithm treats  $t_L$  and  $t_R$  as binary search bounds and modify them correspondingly, until it reaches  $t_R - t_L = 1$ . It keeps the following invariant: the  $(t_L, \lfloor m_2/(n - t_L + 1) \rfloor)$ -split-allocation is LE, and the  $(t_R, \lfloor m_2/(n - t_R + 1) \rfloor)$ -split-allocation is RE. In a single iteration of the binary search, the algorithm takes  $t_M$  in the middle between  $t_L$  and  $t_R$ , checks whether the  $(t_M, \lfloor m_2/(n - t_M + 1) \rfloor)$ -split-allocation is EFX, LE, or RE, and returns the correct allocation, puts  $t_L$  equal to  $t_M$ , or puts  $t_R$  equal to  $t_M$  correspondingly (Lines 8-11).

After the binary search, the algorithm puts  $t := t_L$ . We know that the desired “envy direction change” point has form  $(t, k)$  for some integer  $k$  between  $\lfloor m_2/(n - t + 1) \rfloor$  and 1 (because the successor of  $(t, 1)$  w.r.t. to  $\prec$  is RE). If  $(t, 1)$  is right-envious allocation, then the algorithm performs binary search over  $k$  starting with  $k_L = \lfloor m_2/(n - t + 1) \rfloor$  and  $k_R = 1$  in exactly the same way as in the first binary search (Line 14-18). An only little difference from the first binary search is that  $k_L > k_R$  (instead of  $k_L < k_R$ ), as imposed by definition of  $\prec$ .

If the  $(t, 1)$ -split-allocation is not left-envious, then its either EFX (and forms the correct solution), or  $k = 1$  is viable choice for Lemma 1 (Lines 21-23).

In either of the two cases, the algorithm finds  $(t, k)$  viable for Lemma 1 (or encounters an EFX split allocation and returns before). The algorithm constructs the  $(t, k)$ -reallocation and returns it as a final solution (Line 24). This allocation is EFX+PO by Theorem 4.

Similarly to split allocations, the  $(t, k)$ -reallocation is found in  $\mathcal{O}(1)$  time, since there is at most a constant number of bundles given to contiguous segments of agents.  $\square$

## Envy Directions in Split Allocations

This section is devoted to properties of envy (up to any item) and envy directions in split allocations. Our main goal is to prove Theorem 2 and Theorem 3, that are fundamental to our algorithm.

Before proceeding, we fomulate split allocations numerically for referencing throughout the proof.

**Definition 9** (Numerical definition of split allocations). For  $(t, k) \in \mathcal{T}$ , the  $(t, k)$ -split-allocation  $\{(x_{i,1}, x_{i,2})\}$  is given by

$$x_{i,2} = \begin{cases} 0, & \text{if } i < t, \\ k, & \text{if } i = t, \\ q_2, & \text{if } t < i < n - r_2, \\ q_2 + 1, & \text{if } i > n - r_2, \end{cases}$$

where  $m_2 - k = q_2(n - t) + r_2$  for  $0 \leq r_2 < n - t$ , and, for  $p = \lceil k \cdot v_{t,2} \rceil$ ,

(a) in case of  $m_1 < p(t - 1)$ ,

$$x_{i,1} = \begin{cases} 0, & \text{if } i \geq t, \\ q_1, & \text{if } i \leq t - 1 - r_1, \\ q_1 + 1, & \text{if } i \in [t - r_1, t - 1], \end{cases}$$

where  $m_1 = q_1 \cdot (t - 1) + r_1$  for  $0 \leq r_1 < (t - 1)$ ,

(b) in case of  $m_1 \geq p(t - 1)$ ,

$$x_{i,1} = \begin{cases} 0, & \text{if } i > t, \\ q_1, & \text{if } i = t \text{ and } r_1 = 0, \\ q_1 + 1, & \text{if } i = t \text{ and } r_1 > 0, \\ p + q_1, & \text{if } i \leq \min\{t - r_1, t - 1\}, \\ p + q_1 + 1, & \text{if } i \in [t - r_1 + 1, t - 1], \end{cases}$$

where  $m_1 - p(t - 1) = q_1 \cdot t + r_1$  for  $0 \leq r_1 < t$ .

We start with a series of lemmas. The first simple lemma demonstrates that there can be no envy between agents that go before  $t$  or between two agents that go after  $t$  in a  $(t, k)$ -split-allocation.

**Lemma 2** ( $\star$ ). *Let  $X$  be a  $(t, k)$ -split-allocation for  $(t, k) \in \mathcal{T}$ . If there is envy (up to any item) between agents  $i < j$  in  $X$  for  $i, j \in [n]$ , then  $i \leq t$  and  $j \geq t$ .*

The next lemma restricts any left-envious (LE) split allocation to the case (b) from Definition 9 of split allocations.

**Lemma 3** ( $\star$ ). *Let  $X$  be the  $(t, k)$ -split-allocation for  $(t, k) \in \mathcal{T}$ . If  $X$  is LE, then  $m_1 \geq p \cdot (t - 1)$ , where  $p = \lceil k \cdot v_{t,2} \rceil$ .*

The final lemma in the series explains that  $m_1 \geq p(t - 1)$  also rules out envies between any agent  $i < t$  and agent  $t$ .

**Lemma 4** ( $\star$ ). *Let  $X$  be the  $(t, k)$ -split-allocation for  $(t, k) \in \mathcal{T}$ , and let  $p = \lceil k \cdot v_{t,2} \rceil$ . If  $m_1 \geq p \cdot (t - 1)$ , then for each  $i \in [t - 1]$  there is no envy (up to any item) between agent  $i$  and agent  $t$  in  $X$ .*

We are ready to prove our main result on envy directions in split allocations, Theorem 2. For convenience of the reader, we restate it here first.

**Theorem 2.** For each  $(t, k) \in \mathcal{T}$ , the  $(t, k)$ -split-allocation cannot be LE and RE simultaneously.

*Proof.* Targeting towards a contradiction, assume that there exists  $(t, k) \in \mathcal{T}$  such that the  $(t, k)$ -split-allocation  $X = \{(x_{i,1}, x_{i,2})\}$  is LE and RE simultaneously.

By Lemma 3, we know that  $m_1 \geq p \cdot (t-1)$  holds for  $p = \lceil x_{t,2} \cdot v_{t,2} \rceil$ . In particular,  $X$  comes from case (b) of Definition 9. Then, by Lemma 4 we have that there is no envy (up to any item) between any agent  $i \in [t-1]$  and agent  $t$  in  $X$ . Consequently, if there is an envy between agent  $i$  and agent  $j$  in  $X$ , where  $i < j$ , then  $i \leq t$  and  $j > t$  necessarily holds.

Under the initial assumption that  $X$  is LE and RE simultaneously, we prove two contravensory claims. Recall that  $q_1$  and  $r_1$  from Definition 9 of split allocations are non-negative integers satisfying  $m_1 = q_1 \cdot (t-1) + r_1$ . Similarly,  $q_2$  and  $r_2$  are two positive integers that satisfy  $m_2 - k = q_2 \cdot (n-t) + r_2$ .

The first claim comes from  $X$  being right-envious.

**Claim 2.** Both of the following is true:

- If  $r_1 > 0$ , then  $p + q_1 < q_2 \cdot v_{t,2}$ .
- If  $r_1 = 0$ , then  $v_{t,2} \cdot x_{t,2} + q_1 < q_2 \cdot v_{t,2}$ .

*Proof of Claim 2.* Since  $X$  is right-envious, there is an agent  $i \in [t]$  that envies (up to any item) some agent  $j \in [t+1, n]$ . There are two cases depending on whether  $i < t$  or  $i = t$ .

If  $i < t$ , then agent  $i$  has no items of the second type, and agent  $j$  has no items of the first type. The envy from agent  $i$  to agent  $j$  is equivalent to  $x_{i,1} + 1 \cdot v_{i,2} < x_{j,2} \cdot v_{i,2}$ . From Definition 9 we have that  $x_{i,1} \geq p + q_1$  and  $x_{j,2} \leq q_2 + 1$ . Consequently,  $p + q_1 + v_{i,2} < (q_2 + 1) \cdot v_{i,2}$ . It follows  $p + q_1 < q_2 \cdot v_{i,2} \leq q_2 \cdot v_{t,2}$ , since  $v_{i,2} \leq v_{t,2}$ .

We have that if  $i < t$ , then  $p + q_1 < q_2 \cdot v_{t,2}$ . Since  $p \geq k \cdot v_{t,2} = x_{t,2} \cdot v_{t,2}$ , it also follows  $v_{t,2} \cdot x_{t,2} + q_1 < q_2 \cdot v_{t,2}$ . Therefore, both parts of the claim follow if  $i < t$ .

We now consider the case when agent  $t$  envies (up to any item) agent  $j > t$ . The envy is equivalent to

$$x_{t,1} + (x_{t,2} + 1) \cdot v_{t,2} < x_{j,2} \cdot v_{t,2}.$$

Recall that  $x_{j,2} \leq q_2 + 1$  and rewrite the above as

$$x_{t,1} + x_{t,2} \cdot v_{t,2} < q_2 \cdot v_{t,2}. \quad (2)$$

If  $r_1 = 0$ , then  $x_{t,1} = q_1$ , and (2) becomes

$$x_{t,2} \cdot v_{t,2} + q_1 \leq q_2 \cdot v_{t,2},$$

as required by the first part of the claim statement.

If  $r_1 > 0$ , then  $x_{t,1} = q_1 + 1$ . Then, by  $p < x_{t,2} \cdot v_{t,2} + 1$  and (2) obtain

$$p + q_1 < x_{t,2} \cdot v_{t,2} + 1 + q_1 \leq x_{t,1} + x_{t,2} \cdot v_{t,2} < q_2 \cdot v_{t,2}.$$

This proves the second part of the claim statement. This concludes the proof of the claim.  $\lrcorner$

The second claim comes from  $X$  being left-envious.

**Claim 3.** Both of the following is true:

- If  $r_1 > 0$ , then  $p + q_1 > q_2 \cdot v_{t,2}$ .
- If  $r_1 = 0$ , then  $v_{t,2} \cdot x_{t,2} + q_1 > q_2 \cdot v_{t,2}$ .

*Proof of Claim 3.* Since  $X$  is left-envious, there is an agent  $j > t$  that envies some agent  $i \in [t]$ . There are two cases depending on whether  $i < t$  or  $i = t$ .

Consider first the case when  $i < t$ . The envy from agent  $j$  to agent  $i$  is then equivalent to  $1 + x_{j,2} \cdot v_{j,2} < x_{i,1}$ . We have that  $x_{j,2} \geq q_2$  from definition of  $X$ , and  $v_{j,2} \geq v_{t,2}$  since  $j > t$ . Combining the inequalities, obtain

$$q_2 \cdot v_{t,2} < x_{i,1} - 1. \quad (3)$$

If  $r_1 > 0$ , then  $x_{i,1} \leq p + q_1 + 1$ . Then (3) gives

$$q_2 \cdot v_{t,2} < p + q_1 + 1 - 1,$$

which is essentially the first part of the claim. If  $r_1 = 0$ , then  $x_{i,1} = p + q_1$ . Combining this, (3) and  $v_{t,2} \cdot x_{t,2} > p - 1$  gives

$$q_2 \cdot v_{t,2} < (p - 1) + q_1 < v_{t,2} \cdot x_{t,2} + q_1.$$

The second part is also proved for case  $i < t$ .

We move on to the remaining case  $i = t$ . The envy from agent  $j$  to agent  $t$  is then expressed as  $1 + x_{j,2} \cdot v_{j,2} < x_{t,1} + x_{t,2} \cdot v_{t,2}$ . Similarly to the previous case, rewrite this as

$$q_2 \cdot v_{t,2} < (x_{t,1} - 1) + x_{t,2} \cdot v_{t,2}. \quad (4)$$

If  $r_1 > 0$ ,  $x_{t,1} = q_1 + 1$ . Combining this, (4) and  $p \geq x_{t,2} \cdot v_{t,2}$  gives the first part of the claim. If  $r_1 = 0$ , then  $x_{t,1} = q_1$ . Combining (4) with  $x_{t,1} = q_1$  gives

$$q_2 \cdot v_{t,2} < q_1 - 1 + x_{t,2} \cdot v_{t,2} < q_1 + x_{t,2} \cdot v_{t,2}.$$

The proof of the case  $i = t$  and the whole claim is complete.  $\lrcorner$

Clearly, Claim 2 and Claim 3 are controversial statements that are both true under the initial assumption. The obtained contradiction proves the theorem.  $\square$

While Theorem 2 is fundamental to our split allocation approach, Theorem 3 is pivotal to the binary search over  $\mathcal{T}$ . The proof of Theorem 3 is omitted due to the space constraints and can be found in the appendix.

## Conclusion

In this work, we have made progress in understanding the existence and computability of EFX+PO allocations, establishing that such allocations always exist for two types of goods with positive utilities and developing an efficient  $\mathcal{O}(n \log n + \log m)$ -time algorithm. We are interested in whether these results can be extended to several simple independent cases of a more general open question. It represents critical next steps in the broader research aimed at characterizing when EFX and EFX+PO allocations are guaranteed to exist.

**Open Question 1.** Does an EFX allocation always exist? Does an EFX+PO allocation always exist when all utilities are positive? At least in two settings:

- utility matrix has rank 2,
- three types of goods.

## Acknowledgments

Yuriy Dementiev, Artur Ignatiev and Danil Sagunov: This work was supported by the Ministry of Economic Development of the Russian Federation (IGK 000000C313925P4C0002), agreement №139-15-2025-010.

## References

- Amanatidis, G.; Aziz, H.; Birmpas, G.; Filos-Ratsikas, A.; Li, B.; Moulin, H.; Voudouris, A. A.; and Wu, X. 2023. Fair division of indivisible goods: Recent progress and open questions. *Artificial Intelligence*, 322: 103965.
- Amanatidis, G.; Birmpas, G.; Filos-Ratsikas, A.; Hollender, A.; and Voudouris, A. A. 2021. Maximum Nash welfare and other stories about EFX. *Theoretical Computer Science*, 863: 69–85.
- Amanatidis, G.; Filos-Ratsikas, A.; and Sgouritsa, A. 2024. Pushing the Frontier on Approximate EFX Allocations. In Bergemann, D.; Kleinberg, R.; and Sabán, D., eds., *Proceedings of the 25th ACM Conference on Economics and Computation, EC 2024, New Haven, CT, USA, July 8-11, 2024*, 1268–1286. ACM.
- Amanatidis, G.; Markakis, E.; and Ntokos, A. 2020. Multiple birds with one stone: Beating 1/2 for EFX and GMMS via envy cycle elimination. *Theor. Comput. Sci.*, 841: 94–109.
- Aziz, H.; Li, B.; Moulin, H.; and Wu, X. 2022. Algorithmic fair allocation of indivisible items: a survey and new questions. *SIGecom Exch.*, 20(1): 24–40.
- Berger, B.; Cohen, A.; Feldman, M.; and Fiat, A. 2022. Almost Full EFX Exists for Four Agents. In *Thirty-Sixth AAAI Conference on Artificial Intelligence, AAAI 2022, Thirty-Fourth Conference on Innovative Applications of Artificial Intelligence, IAAI 2022, The Twelfth Symposium on Educational Advances in Artificial Intelligence, EAAI 2022 Virtual Event, February 22 - March 1, 2022*, 4826–4833. AAAI Press.
- Budish, E. 2011. The Combinatorial Assignment Problem: Approximate Competitive Equilibrium from Equal Incomes. *Journal of Political Economy*, 119(6): 1061 – 1103.
- Caragiannis, I.; Gravin, N.; and Huang, X. 2019. Envy-Freeness Up to Any Item with High Nash Welfare: The Virtue of Donating Items. In Karlin, A. R.; Immorlica, N.; and Johari, R., eds., *Proceedings of the 2019 ACM Conference on Economics and Computation, EC 2019, Phoenix, AZ, USA, June 24-28, 2019*, 527–545. ACM.
- Caragiannis, I.; Kurokawa, D.; Moulin, H.; Procaccia, A.; Shah, N.; and Wang, J. 2019. The Unreasonable Fairness of Maximum Nash Welfare. *ACM Transactions on Economics and Computation*, 7: 1–32.
- Chaudhury, B. R.; Garg, J.; and Mehlhorn, K. 2024. EFX Exists for Three Agents. *J. ACM*, 71(1): 4:1–4:27.
- Chaudhury, B. R.; Kavitha, T.; Mehlhorn, K.; and Sgouritsa, A. 2021. A Little Charity Guarantees Almost Envy-Freeness. *SIAM J. Comput.*, 50(4): 1336–1358.
- Foley, D. K. 1966. *Resource allocation and the public sector*. Yale University.
- Garg, J.; and Murhekar, A. 2023. Computing fair and efficient allocations with few utility values. *Theoretical Computer Science*, 962: 113932.
- Ghosal, P.; HV, V. P.; Nimbhorkar, P.; and Varma, N. 2025. (Almost Full) EFX for Three (and More) Types of Agents. In Walsh, T.; Shah, J.; and Kolter, Z., eds., *AAAI-25, Sponsored by the Association for the Advancement of Artificial Intelligence, February 25 - March 4, 2025, Philadelphia, PA, USA*, 13889–13896. AAAI Press.
- Gorantla, P.; Marwaha, K.; and Velusamy, S. 2023. Fair allocation of a multiset of indivisible items. In Bansal, N.; and Nagarajan, V., eds., *Proceedings of the 2023 ACM-SIAM Symposium on Discrete Algorithms, SODA 2023, Florence, Italy, January 22-25, 2023*, 304–331. SIAM.
- Hosseini, H.; Sikdar, S.; Vaish, R.; and Xia, L. 2021. Fair and Efficient Allocations under Lexicographic Preferences. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*, 5472–5480. AAAI Press.
- Lipton, R. J.; Markakis, E.; Mossel, E.; and Saberi, A. 2004. On Approximately Fair Allocations of Indivisible Goods. In *Proceedings of the 5th ACM Conference on Electronic Commerce, EC '04*, 125–131. New York, NY, USA: Association for Computing Machinery. ISBN 1581137710.
- Mahara, R. 2023. Existence of EFX for two additive valuations. *Discrete Applied Mathematics*, 340: 115–122.
- Mahara, R. 2024. Extension of Additive Valuations to General Valuations on the Existence of EFX. *Math. Oper. Res.*, 49(2): 1263–1277.
- Nguyen, T. T.; and Rothe, J. 2023. Complexity Results and Exact Algorithms for Fair Division of Indivisible Items: A Survey. In *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence, IJCAI 2023, 19th-25th August 2023, Macao, SAR, China*, 6732–6740. ijcai.org.
- Plaut, B.; and Roughgarden, T. 2020. Almost Envy-Freeness with General Valuations. *SIAM J. Discret. Math.*, 34(2): 1039–1068.
- Prakash HV, V.; Ghosal, P.; Nimbhorkar, P.; and Varma, N. 2025. EFX Exists for Three Types of Agents. In Ashlagi, I.; and Roth, A., eds., *Proceedings of the 26th ACM Conference on Economics and Computation, EC 2025, Stanford University, Stanford, CA, USA, July 7-10, 2025*, 101–128. ACM.