

Best-of-Both-Worlds Fair Allocation of Indivisible and Mixed Goods*

Xiaolin Bu¹, Zihao Li², Shengxin Liu³, Xinhang Lu⁴, and Biaoshuai Tao¹

¹Shanghai Jiao Tong University, {lin_bu, bstao}@sjtu.edu.cn

²Nanyang Technological University, zihao004@e.ntu.edu.sg

³Harbin Institute of Technology, Shenzhen, sxliu@hit.edu.cn

⁴UNSW Sydney, xinhang.lu@unsw.edu.au

Abstract

We study the problem of fairly allocating either a set of indivisible goods or a set of mixed divisible and indivisible goods (i.e., mixed goods) to agents with additive utilities, taking the best-of-both-worlds perspective of guaranteeing fairness properties both ex-ante and ex-post. The ex-post fairness notions considered in this paper are relaxations of envy-freeness, specifically, EFX for indivisible-goods allocation, and EFM for mixed-goods allocation. For two agents, we show that there is a polynomial-time randomized algorithm that achieves ex-ante envy-freeness and ex-post EFX / EFM simultaneously. For n agents with bi-valued utilities, we show there exist randomized allocations that are (i) ex-ante proportional and ex-post EFM, and (ii) ex-ante envy-free, ex-post EFX, and ex-post fractionally Pareto optimal.

1 Introduction

Fair division studies the problem of fairly allocating scarce resources among agents with heterogeneous preferences over the resources. It is a fundamental problem in society, with real-world applications such as divorce settlement, rent division, course allocation, and more. Two classic fairness notions in the literature are *envy-freeness* (*EF*) and *proportionality* (*PROP*). An allocation is said to be envy-free if each agent values her own bundle weakly better than any other bundle in the allocation, and proportional if every agent gets a bundle of value at least $1/n$ times her value for the entire resources, where n is the number of agents.

Our focus in this paper is on the settings of allocating *indivisible* goods as well as a mixture of both divisible and indivisible goods (henceforth referred to as *mixed* goods) when agents have additive utilities, which have received significant attention in recent years [Amanatidis et al., 2023; Liu et al., 2024; Suksompong, 2021, 2025]. Despite being desirable properties, neither envy-freeness nor proportionality can always be satisfied when (deterministically) allocating indivisible or mixed goods among the agents.

To circumvent the issue, relaxations of the notions have been proposed and studied. For instance, when allocating indivisible goods, *envy-freeness up to any good* (*EFX*) (resp., *envy-freeness up to one good* (*EF1*)) requires that an agent's envy towards another agent should be eliminated

*A preliminary version appears in *Proceedings of the 20th Conference on Web and Internet Economics (WINE)* [Bu et al., 2024].

after the hypothetical removal of any (resp., some) good from the latter agent’s bundle [Budish, 2011; Caragiannis et al., 2019b]. While the existence of EFX allocations is only known in special cases (cf. Section 1.2), the weaker notion of EF1 can always be satisfied, even for any number of agents with arbitrary monotonic utilities [Lipton et al., 2004]. With mixed goods, Bei et al. [2021a] proposed *envy-freeness for mixed goods (EFM)*, which generalizes both envy-freeness and EF1 in the following sense: An agent is envy-free towards any agent whose bundle contains some divisible goods and EF1 towards the rest. They showed that an EFM allocation always exists for any number of agents with additive utilities.

An alternative and common method to achieve fairness is through randomization [see, e.g., Abdulkadiroğlu and Sönmez, 1998; Akbarpour and Nikzad, 2020; Bogomolnaia and Moulin, 2001; Budish et al., 2013; Gözl et al., 2024]. With the power of randomization, envy-freeness and proportionality can both be easily and trivially achieved by giving all goods to a single agent uniformly at random. The realized allocation, however, is patently unfair since all agents but one are left empty-handed.

Aziz et al. [2024a] timely introduced the *best-of-both-worlds (BoBW)* approach, which combines the two aforementioned methods with the goal of constructing a randomized allocation (i.e., a lottery over deterministic allocations) that is exactly fair ex-ante (before the randomness is realized) and approximately fair ex-post (after the randomness is realized). They showed that ex-ante envy-freeness and ex-post EF1 can be simultaneously achieved when agents have additive utilities. In this paper, our goal is to strengthen the ex-post fairness guarantee to EFX when allocating indivisible goods, and, for the first time, extend the study of best-of-both-worlds fairness to the mixed-goods setting.

1.1 Our Results

As is common in the literature of fair division, we assume that agents have additive utilities when allocating either indivisible or mixed goods.

In Section 3, we focus on two agents. While the work of Feldman et al. [2024] showed that there always exists a randomized allocation that is ex-ante EF and ex-post EFX, we improve upon the result by devising a *polynomial-time* algorithm to compute such a randomized allocation.¹ Built upon this result, we then show with mixed goods, a randomized allocation that is simultaneously ex-ante EF and ex-post EFM can be found in polynomial time.

Next, we consider the n -agent cases, with Sections 4 and 5 focusing on mixed-goods and indivisible-goods settings, respectively. In both cases, we assume that agents have *bi-valued* utilities, i.e., each agent’s utility for each good belongs to one of two possible values.²

In Section 4, we devise a polynomial-time algorithm to compute an integral allocation sampled from a randomized allocation that is ex-ante proportional and ex-post EFM. Our result on the compatibility between EFM and ex-ante fairness notions adds to the growing literature revolved around EFM when allocating mixed resources [Bhaskar et al., 2021; Li et al., 2024a, 2023, 2024b; Nishimura and Sumita, 2023].

In Section 5, we devise a polynomial-time algorithm to compute an integral allocation sampled from a randomized allocation that is ex-ante EF, ex-post EFX, and ex-post fPO.³ This generalizes

¹An independent work of Garg and Sharma [2024] also gives a polynomial-time algorithm to achieve the same goal by using a different approach. In addition to ex-ante EF and ex-post EFX, the method of Garg and Sharma [2024] can additionally achieve ex-post 4/5-approximation of the *maximin share*.

²Binary utilities (the two possible values are 0 and 1) are special cases.

³Ex-post *fractionally Pareto optimality (fPO)*, which can be found in Definition 2.3, is a stronger notion of economic efficiency than ex-post Pareto optimality (PO).

multiple results known in the literature. For instance, the compatibility between ex-ante EF, ex-post EFX, and ex-post PO was only known for binary utilities [Babaioff et al., 2021; Halpern et al., 2020].⁴ For bi-valued utilities, we only knew the compatibility of ex-post notions between EFX and PO [Amanatidis et al., 2021] as well as between EFX and fPO [Garg and Murhekar, 2023].

1.2 Related Work

The fair allocation of indivisible goods has received extensive attention in the past decades [Amanatidis et al., 2023; Suksompong, 2021, 2025]. Liu et al. [2024] overviewed the recent developments of mixed-goods allocations. Below, we first discuss EFX existence results, followed by an overview of best-of-both-worlds fairness in fair division.

EFX Existence The existence of EFX allocations remains largely open, though progress has been made in special cases where the number of agents or agents’ utility functions are restricted. An EFX allocation always exists for two agents with general utilities [Plaut and Roughgarden, 2020], for three agents [Akrami et al., 2023a; Chaudhury et al., 2024], as well as for any number of agents with restricted utilities such as being identical [Plaut and Roughgarden, 2020], binary submodular [Babaioff et al., 2021] or more general [Bu et al., 2023], bi-valued [Amanatidis et al., 2021; Garg and Murhekar, 2023], or when there are two types of agents and agents of the same type have identical additive utilities [Mahara, 2023]. There have been lines of work focusing on approximately-EFX allocations [Akrami et al., 2023a; Amanatidis et al., 2020], partial allocations that are EFX [Berger et al., 2022; Caragiannis et al., 2019a; Chaudhury et al., 2021], or a combination of both [Chaudhury et al., 2023].

BoBW Fairness In addition to the compatibility result between ex-ante EF and ex-post EF1 mentioned above, Aziz et al. [2024a] showed several impossibility results regarding achieving BoBW fair and economically efficient allocations. For agents with weighted *entitlements*, ex-ante weighted envy-freeness (WEF) is compatible with ex-post weighted transfer envy-freeness up to one good, but not compatible with any stronger ex-post WEF relaxation [Aziz et al., 2023a; Hoefler et al., 2023]. For agents with subadditive utilities, ex-ante $\frac{1}{2}$ -EF, ex-post $\frac{1}{2}$ -EFX and ex-post EF1 can be achieved simultaneously [Feldman et al., 2024]. Best-of-both-worlds fairness has also been explored for fair-share-based notions like proportionality and the *maximin share* (MMS) guarantee for agents with additive [Akrami et al., 2024; Babaioff et al., 2022] or fractionally subadditive utilities [Akrami et al., 2023b]. Babaioff et al. [2022] showed ex-ante proportionality and ex-post $\frac{1}{2}$ -MMS are compatible. The ex-post MMS approximation ratio was improved in [Akrami et al., 2024], at the cost of weakening ex-ante fairness guarantees.

Slightly further afield, the BoBW paradigm has also been applied to the contexts of collective choice such as committee voting [Aziz et al., 2023b; Suzuki and Vollen, 2024] and participatory budgeting [Aziz et al., 2024b].

2 Preliminaries

For each positive integer t , let $[t] := \{1, 2, \dots, t\}$. Denote by $N = [n]$ the set of n agents to whom we allocate resources. In this paper, we consider both indivisible-goods and mixed-goods settings. Let $M \cup D$ be the set of mixed goods, where $M = \{g_1, g_2, \dots, g_m\}$ is the set of m indivisible goods

⁴The result of Babaioff et al. [2021] works for *binary submodular* (also known as *matroid-rank*) utilities.

and $D = \{d_1, d_2, \dots, d_{\bar{m}}\}$ is the set of \bar{m} homogeneous divisible goods. When $D = \emptyset$, we are concerned with the indivisible-goods setting.

Allocations A fractional allocation of mixed goods $M \cup D$ to the agents in N is specified by a non-negative $n \times (m + \bar{m})$ matrix $\mathbf{X} = (X_{ig})_{i \in N, g \in M \cup D}$ such that for each $g \in M \cup D$, $\sum_{i \in N} X_{ig} = 1$; here, $X_{ig} \in [0, 1]$ denotes the fraction of good g assigned to agent i . The i -th row X_i of the matrix denotes the goods allocated to agent i in the fractional allocation. When we simply say “an allocation”, it will mean a fractional allocation, unless otherwise clear from the context.

A fractional allocation \mathbf{X} is *integral* if $X_{ig} \in \{0, 1\}$ for every $i \in N$ and every indivisible good $g \in M$. Given an integral allocation \mathbf{X} , denote by $M_i := \{g \in M \mid X_{ig} = 1\}$ the set of indivisible goods assigned to agent i , and let the \bar{m} -dimensional vector $\mathbf{x}_i = (x_{i1}, x_{i2}, \dots, x_{i\bar{m}}) := (X_{ig})_{g \in D}$ represent the divisible goods received by agent i . We will refer to $A_i = (M_i, \mathbf{x}_i)$ as the *bundle* of mixed goods allocated to agent i . The integral allocation is written as $\mathcal{A} = (A_i)_{i \in N}$. We slightly abuse notation by also using \mathcal{A} as an $n \times (m + \bar{m})$ -matrix representation of the integral allocation.

A *randomized allocation* is a probability distribution over integral allocations and specified by a set of $s \in \mathbb{N}$ tuples $\{(p_j, \mathcal{A}_j)\}_{j \in [s]}$, where $p_j \in [0, 1]$, $\sum_{j \in [s]} p_j = 1$, and \mathcal{A}_j is an integral allocation implemented with probability p_j . A randomized allocation $\{(p_j, \mathcal{A}_j)\}_{j \in [s]}$ is called an *implementation* of (or, interchangeably, *implements*) a fractional allocation \mathbf{X} if $\mathbf{X} = \sum_{j \in [s]} p_j \cdot \mathcal{A}_j$.

Utilities Each agent $i \in N$ is associated with an *additive* utility function u_i , where $u_i(g) \geq 0$ denotes the agent’s utility for fully receiving good $g \in M \cup D$. We say that agents have *bi-valued* utilities if for each $i \in N$ and $g \in M \cup D$, $u_i(g) \in \{a, b\}$, where $0 \leq a < b$ are distinct, non-negative real values.⁵ When $a = 0$ and $b = 1$, agents are said to have *binary* utilities. Given an allocation \mathbf{X} , the utility of agent i is given by $u_i(\mathbf{X}) = u_i(X_i) = \sum_{g \in M \cup D} X_{ig} \cdot u_i(g)$.

Solution Concepts We now introduce fairness notions. The first two concepts—envy-freeness [Foley, 1967; Varian, 1974] and proportionality [Steinhaus, 1948]—are defined for allocations, fractional or integral. An allocation \mathbf{X} is said to satisfy *envy-freeness* (EF) if for any pair of agents $i, j \in N$, $u_i(X_i) \geq u_i(X_j)$, and *proportionality* (PROP) if for any agent $i \in N$, $u_i(X_i) \geq \frac{u_i(M \cup D)}{n}$. Our next notions are compelling relaxations of envy-freeness when concerning integral allocations of indivisible goods.

Definition 2.1 (EF1 [Budish, 2011; Lipton et al., 2004] and EFX [Caragiannis et al., 2019b; Plaut and Roughgarden, 2020]). An integral allocation $(M_i)_{i \in N}$ of indivisible goods M is said to satisfy

- *envy-freeness up to one good* (EF1) if for any pair of agents $i, j \in N$ such that $M_j \neq \emptyset$, we have $u_i(M_i) \geq u_i(M_j \setminus \{g\})$ for some $g \in M_j$;
- *envy-freeness up to any good* (EFX) if for any pair of agents $i, j \in N$ such that $M_j \neq \emptyset$, we have $u_i(M_i) \geq u_i(M_j \setminus \{g\})$ for all $g \in M_j$.

With mixed goods, the following notion combines and generalizes both envy-freeness and EF1.

Definition 2.2 (EFM [Bei et al., 2021a]). An integral allocation $(A_i)_{i \in N}$ of mixed goods $M \cup D$ is said to satisfy *envy-freeness for mixed goods* (EFM) if for any pair of agents $i, j \in N$,

- if agent j ’s bundle consists of only indivisible goods (i.e., $\mathbf{x}_j = \mathbf{0}$) and $M_j \neq \emptyset$, there exists some $g \in M_j$ such that $u_i(A_i) \geq u_i(M_j \setminus \{g\})$;

⁵We will later use an agent’s “large good” (resp., “small good”) to refer to a good of value b (resp., a) to the agent.

- otherwise, $u_i(A_i) \geq u_i(A_j)$.

Finally, we introduce an economic efficiency notion of importance in the context of fair division.

Definition 2.3 (fPO). An integral allocation $(M_i)_{i \in N}$ of indivisible goods M is said to satisfy *fractionally Pareto optimal (fPO)* if there is no fractional allocation \mathbf{Y} of indivisible goods M such that $u_i(Y_i) \geq u_i(M_i)$ for all agents $i \in N$ and $u_j(Y_j) > u_j(M_j)$ for some agent $j \in N$.

We say a randomized allocation satisfies a property ex-ante (resp., ex-post) if the fractional allocation it implements (resp., every integral allocation in its support) satisfies the property. This paper concerns the problem of designing randomized algorithms that simultaneously achieve desirable properties both ex-ante and ex-post. Our algorithms do not explicitly output the desired randomized allocation and instead *sample* integral allocations from their supports.

3 Two Agents

In this section, we study the best-of-both-worlds fair allocation of indivisible and mixed goods for two agents. With indivisible goods, [Feldman et al. \[2024\]](#) showed the existence of a randomized allocation that is ex-ante EF and ex-post EFX. We improve upon their result by presenting a *polynomial-time* algorithm to identify such an allocation. We then leverage this algorithm to provide a randomized allocation of mixed goods that is both ex-ante EF and ex-post EFM in polynomial time.

An independent work of [Garg and Sharma \[2024\]](#) gives a fundamentally different polynomial-time algorithm to achieve the same goal, ex-ante EF and ex-post EFX. In addition, their algorithm can achieve the extra fairness guarantee of ex-post $\frac{4}{5}$ -MMS.

In Proposition A.2 and Lemma A.1 of [Feldman et al. \[2024\]](#), the requirement for ex-post EFX involves initially identifying two allocations, denoted as \mathcal{A}^1 and \mathcal{A}^2 , which minimize the difference between two bundles under utility functions u_1 and u_2 respectively. The cut-and-choose protocol is then employed to determine the owner of each bundle in both allocations. The minimization of differences ensures that, for each agent $i \in [2]$, the loss in allocation \mathcal{A}^i can be bounded by the gain in allocation \mathcal{A}^{3-i} , thereby ensuring ex-ante EF. Their mechanism involves an NP-hard step of finding an allocation that minimizes the difference. In contrast, our focus is solely on identifying two corresponding EFX allocations under utility functions u_1 and u_2 , ensuring that the loss in \mathcal{A}^i is bounded by the gain in \mathcal{A}^{3-i} for each agent i . Motivated by this, we present our algorithm, whose pseudocode can be found in Algorithm 3 in Appendix A.

In Algorithm 3, we incorporate a subroutine called `LocalSearch` (lines 12 to 18), which, given an arbitrary allocation (A, B) and a utility function u , returns an integral EFX allocation with respect to u .⁶ The following lemma shows this subroutine produces in polynomial time an EFX allocation with a weakly smaller utility difference between its two bundles compared to the provided allocation (A, B) . Its proof, along with all other omitted proofs, can be found in the appendices.

⁶While our `LocalSearch` technique and the proof technique used by [Plaut and Roughgarden \[2020\]](#) to show their Theorem 4.2 share certain similarities, as we both update the current non-EFX allocation by moving some good from a bundle of higher value to a bundle of lower value, there are differences as well, mainly because we consider different utility classes. For instance, [Plaut and Roughgarden](#) showed the *leximin++ solution* is EFX for n agents with general but identical utilities. In addition to bundles' values, the *leximin++ solution* also takes into account of bundles' sizes. But since we focus on two agents with additive utilities, our `LocalSearch` only updates allocations based on bundles' values.

Lemma 3.1. $\text{LocalSearch}(A, B, u)$ returns in polynomial time an integral EFX allocation (A', B') under utility function u with $|u(B') - u(A')| \leq |u(B) - u(A)|$.

By leveraging the subroutine, we now describe our main procedure (lines 1 to 11). For each agent $i \in [2]$, we maintain an allocation \mathcal{A}^i which is an integral EFX allocation under u_i . If one of them is an envy-free allocation under the utility profile (u_1, u_2) (lines 2 to 3), we return this allocation directly. Otherwise, either

- there exists an allocation \mathcal{A}^i which has a smaller utility difference over two bundles than that in \mathcal{A}^{3-i} under u_{3-i} , we update \mathcal{A}^{3-i} from \mathcal{A}^i by using LocalSearch (lines 4-10), or
- we find the desired ex-ante EF allocation $\{(0.5, \mathcal{A}^1), (0.5, \mathcal{A}^2)\}$ (line 11).

Theorem 3.2. In the indivisible-goods setting, Algorithm 3 returns an ex-ante EF and ex-post EFX allocation in polynomial time for two agents with additive utilities.

In the context of the mixed-goods setting, Algorithm 3 can be employed to identify an allocation that is both ex-ante EF and ex-post EFM. Specifically, we first merge all divisible goods in D to a single indivisible good d with an equal total utility and then execute Algorithm 3 on $M \cup \{d\}$. If the output is a single integral allocation, it is an EF (thus EFM) allocation. Otherwise, if there exists an agent $i \in [2]$ such that the good d is in bundle A_2^i (the bundle with the larger utility), we can transfer a fraction of d to A_1^i to reach EF. If no such agent exists, since all divisible goods are consistently in the bundle with the smaller utility, EFM can be reduced to EF1, which is then satisfied by the ex-post EFX property.

Theorem 3.3. In the mixed-goods setting, an ex-ante EF and ex-post EFM allocation can be found in polynomial time for two agents with additive utilities.

We remark that the above procedure can also ensure a stronger ex-post fairness property called *envy-freeness up to any indivisible good for mixed goods (EFXM)* [Bei et al., 2021a; Nishimura and Sumita, 2023], which replaces the adopted EF1 criteria by the EFX criteria when comparing to the bundle containing only indivisible goods.

4 BoBW Fairness with Mixed Goods: Ex-Ante PROP + Ex-Post EFM

In this section, we study the best-of-both-worlds fairness in the mixed-goods setting where agents have bi-valued utilities over the set of divisible and indivisible goods. Our main result is the following:

Theorem 4.1. In the mixed-goods setting where agents have bi-valued utilities, there exists an algorithm which can compute in polynomial time an integral allocation sampled from a randomized allocation that is ex-ante proportional and ex-post EFM.

4.1 Technical Overview

We have devised the following techniques specifically for BoBW fairness with mixed goods.

Baseline comparison. This technique is designed to show our randomized allocation satisfies ex-ante proportionality. Fix any agent i , we construct a *baseline allocation* $\mathcal{B}^i = (B_1^i, \dots, B_n^i)$ based on u_i . We partition the outcomes of the randomized allocations to n events $\mathcal{E}_1, \dots, \mathcal{E}_n$ and show that, for each $k = 1, \dots, n$, the utility for agent i under each outcome in the event \mathcal{E}_k is at least $u_i(B_k^i)$. The expected utility of agent i is then at least $\sum_{k=1}^n \Pr(\mathcal{E}_k) \cdot u_i(B_k^i)$. If we can show \mathcal{B}^i satisfies

$$\sum_{k=1}^n \Pr(\mathcal{E}_k) \cdot u_i(B_k^i) \geq \frac{1}{n} \cdot u_i(M \cup D), \quad (1)$$

then the ex-ante proportionality for agent i is proved.

To apply this technique, we need to carefully design the baseline allocation \mathcal{B}^i for each agent i and the partition to the n events such that Equation (1) holds. A natural idea is to design the partition of the event space with $\Pr(\mathcal{E}_1) = \dots = \Pr(\mathcal{E}_n) = \frac{1}{n}$ such that Equation (1) holds with equality for any baseline allocation \mathcal{B}^i . This technique is first illustrated in Section 4.3 where $m \leq n$. In the later part with general m , more sophisticated baseline allocations are constructed.

Minimal unmatchable group. Our algorithm starts by allocating indivisible goods, and the divisible goods are allocated by a water-filling process [Bei et al., 2021a]. To guarantee BoBW fairness, the indivisible goods must be allocated carefully in order to satisfy some properties that enable the application of the baseline comparison technique described above. We have interpreted the allocation problem as a matching problem in bipartite graphs and identified a key structure, *minimal unmatchable groups*, that is crucial for BoBW fairness. Many other techniques in the bipartite graph matching problem such as *augmenting paths* are also applied in our result. See Section 4.4 for more details.

4.2 Preparations

First of all, we justify that we can assume without loss of generality that there is only one divisible good, denoted as d . We also let $u_i(d) := u_i(D) = \sum_{j=1}^{\bar{m}} u_i(d_j)$ for each $i \in N$. Specifically, whenever we say that an ϵ portion of d is allocated to an agent i , we refer to the scenario of allocating an ϵ portion of each divisible good $d_1, \dots, d_{\bar{m}}$ to the agent, i.e., $x_{i1} = \dots = x_{i\bar{m}} = \epsilon$. In this way, the set of \bar{m} divisible goods can be treated as the single divisible good d . In the remainder of this section, we thus assume that there is only one divisible good $D = \{d\}$.

Given an EF1 allocation of indivisible goods, Algorithm 1 of Bei et al. [2021a] specifies a way to allocate the divisible goods and finally obtains an EFM allocation of the mixed goods. We refer to this process of allocating divisible goods on top of an EF1 allocation of indivisible goods as the *water-filling* process. During the water-filling process, agents may swap their bundles; however, no bundle will be split. This is formally stated in the following lemma.

Lemma 4.2 (Bei et al. [2021a]). *Given an EF1 allocation $(M_i)_{i \in N}$ of indivisible goods M , an EFM allocation $((M_{\pi(1)}, \mathbf{x}_1), \dots, (M_{\pi(n)}, \mathbf{x}_n))$ can be computed in polynomial time, where π is a permutation of $[n]$.*

4.3 A Special Case where $m \leq n$

We first consider the case $m \leq n$, i.e., the number of indivisible goods is at most the number of agents. This result will be used in later parts. Note, utility functions are not required to be bi-valued.

Algorithm 1: An ex-ante PROP and ex-post EFM randomized allocation when $m \leq n$

Input: Agents N , mixed goods $M \cup D$, and agents' utility functions.

- 1 $\forall i \in [n], A_i \leftarrow \emptyset$
- 2 Let π be a uniformly random permutation of N .
- 3 **foreach** agent i according to the order π **do** // Round-Robin.
- 4 $g^* \leftarrow \arg \max_{g \in M} u_i(g)$
- 5 $A_i \leftarrow A_i \cup \{g^*\}, M \leftarrow M \setminus \{g^*\}$
- 6 Execute the water-filling process. // See Lemma 4.2.
- 7 **return** Allocation \mathcal{A}^π

Our algorithm is shown in Algorithm 1. The Round-Robin algorithm is adopted in the first step to allocate the indivisible goods where the order of the agents π is sampled uniformly at random. The water-filling process is then executed to allocate divisible goods and obtain allocation \mathcal{A}^π . The required randomized allocation we find is $\{(\frac{1}{n!}, \mathcal{A}^\pi)\}$ and is denoted by \mathcal{R} .

Theorem 4.3. *In the mixed-goods setting where agents have additive utilities and $m \leq n$, Algorithm 1 computes in polynomial time an integral allocation sampled from a randomized allocation that is ex-ante proportional and ex-post EFM.*

Proof. We can observe Algorithm 1 runs in polynomial time. By the property of Round-Robin, the allocation after the `foreach`-loop is EF1. Then, by Lemma 4.2, the final output allocation is EFM; hence, the randomized allocation is ex-post EFM.

To show the randomized allocation is ex-ante proportional, a key observation is that, for any fixed partition (X_1, \dots, X_n) of $M \cup D$, if an agent has a probability of $\frac{1}{n}$ to receive each bundle X_i , then her expected utility is exactly $\frac{u_i(M \cup D)}{n}$. As the permutation of the agents is generated uniformly at random in line 2, the probability that an agent i is ranked k -th in the order is $\frac{1}{n}$. Our goal is to define a *baseline allocation* $\mathcal{B}^i = (B_1^i, \dots, B_n^i)$ for each agent $i \in N$ where the value of the bundle that agent i receives in the actual allocation when she is ranked k -th is no less than the k -th bundle in the baseline allocation (i.e., B_k^i) regardless of the permutation of other agents.

Given a permutation π , we use $A(\pi, k)$ to denote the bundle allocated to the agent ranked k -th in the allocation output by Algorithm 1. Consider an arbitrary permutation π where agent i is ranked k -th (i.e., $\pi(k) = i$), we will show that $u_i(A(\pi, k)) \geq u_i(B_k^i)$ where $\mathcal{B}^i = (B_1^i, \dots, B_n^i)$ is the baseline allocation for agent i defined below.

Definition 4.4 (Baseline allocation). The *baseline allocation* \mathcal{B}^i for agent i is obtained by

1. letting agent i partition the indivisible goods M into n bundles using the Round-Robin algorithm (in particular, when $m \leq n$, the t -th bundle (i.e., B_t^i) contains the t -th preferred good for $t \leq m$ while the t -th bundle is empty for $t > m$), and
2. executing the water-filling process for the divisible goods according to agent i 's utility to obtain an EFM allocation for the utility profile (u_i, u_i, \dots, u_i) .

The following properties of \mathcal{B}^i are straightforward. First, each B_j^i contains at most one indivisible good (i.e., the j -th preferred indivisible good of agent i if $j \leq m$, and no indivisible good if $j > m$). Second, to guarantee EFM, the bundles containing some fraction of the divisible good must have the same value under $u_i(\cdot)$. Let x be that value. Third, due to EFM, $u_i(B_j^i) \geq x$ for each $j \in [n]$.

Next, we show that $u_i(A(\pi, k)) \geq u_i(B_k^i)$ for any permutation π with $\pi(k) = i$. It holds trivially when $B_k^i = \emptyset$, or when B_k^i only contains one indivisible good as agent i will receive at least her k -th preferred good during the Round-Robin phase of Algorithm 1. When B_k^i contains divisible goods, we need to show that $u_i(A(\pi, k)) \geq x$ (recall that x denotes the value of the bundles with divisible goods in \mathcal{B}^i). Suppose this is not the case, and $u_i(A_i) < x$ in the actual output allocation $\mathcal{A} = (A_1, \dots, A_n)$ where $A_i = A(\pi, k)$. Let \mathcal{I} be the set of indices of bundles in $\{A_1, \dots, A_n\}$, each of which is envied by agent i . It is worth noting that due to that allocation \mathcal{A} satisfies EFM, those bundles do not contain any divisible good. It is also easy to see $x > u_i(A_i) \geq u_i(A_j)$ for all $j \notin \mathcal{I}$. Hence, we reach the following contradiction:

$$u_i(M \cup D) = \sum_{j \in \mathcal{I}} u_i(A_j) + \sum_{j \notin \mathcal{I}} u_i(A_j) < \sum_{j=1}^{|\mathcal{I}|} u_i(B_j^i) + (n - |\mathcal{I}|) \cdot x \leq \sum_{j=1}^n u_i(B_j^i) = u_i(M \cup D). \quad (2)$$

Finally, we have

$$u_i(\mathcal{R}) = \sum_{k=1}^n \sum_{\pi: \pi(k)=i} \frac{u_i(A(\pi, k))}{n!} \geq \sum_{k=1}^n \sum_{\pi: \pi(k)=i} \frac{u_i(B_k^i)}{n!} = \sum_{k=1}^n \frac{u_i(B_k^i)}{n} = \frac{u_i(M \cup D)}{n},$$

which implies the allocation satisfying ex-ante proportionality. \square

Note that the above algorithm can hardly be generalized to the case with more than n indivisible goods. Suppose we define the baseline allocation for each agent in the same way and B_k^i contains divisible goods, then Equation (2) may fail. In more detail, bundle A_j , where $j \in \mathcal{I}$ may contain more than one indivisible good. Hence, it may be that $\sum_{j \in \mathcal{I}} u_i(A_j) > \sum_{j=1}^{|\mathcal{I}|} u_i(B_j^i)$. As a result, Equation (2) does not necessarily hold, leading to the possibility of $u_i(A(\pi, k)) = u_i(A_i) < u_i(B_k^i)$.

4.4 General m with Bi-Valued Utilities

We now proceed to the general case with arbitrary numbers of goods. We only provide the high-level ideas here, and the details are deferred to Appendix B.

Our algorithm allocates the indivisible goods iteratively and then allocates the divisible goods by the water-filling process. At each iteration, we attempt to allocate each agent a large good of value b . If we construct a bipartite graph $G = (U, V, E)$ where U denotes the set of agents, V denotes the set of indivisible goods, and an edge represents that an agent has value b to a good, then we can find a maximum matching in G at each iteration. If a matching of size n (i.e., a perfect matching) is found, we allocate each agent a “large good” and move on to the next iteration.

At some iteration, a perfect matching may no longer exist. We identify agents $Z \subseteq U$ such that

1. Z cannot be fully matched to large goods; i.e., the Hall’s condition fails for Z : $|\Gamma(Z)| < |Z|$, and
2. the remaining agents $U \setminus Z$ can be fully matched; in addition, for each indivisible good that has value b to an agent in $U \setminus Z$, it has value a to any agent in Z .

Intuitively speaking, agents in Z are “mostly finished” with their large goods, while a perfect matching may continue to exist in the future iterations for agents in $U \setminus Z$. We call a minimal set of agents Z satisfying the above requirements a *minimal unmatchable group*. Its precise definition and the algorithm to find such a group are presented in Appendix B.1.

At each iteration, agents Z and indivisible goods $\Gamma(Z)$ are temporarily removed. We find a perfect matching for agents in $U \setminus Z$, and allocate a “small good” to each agent in Z . In future iterations, we recursively handle $U \setminus Z$. As a result, more and more agents (and the corresponding “neighboring goods”) are removed. For agents that remain, we find a perfect matching and correspondingly allocate each of them a “large good”; for agents that are removed, we allocate each of them a “small good”. Thus, in each iteration, exactly n goods are allocated and each agent receives exactly one good. We stop when the number of the remaining indivisible goods is less than n .

Now, the number of unallocated indivisible goods is at most $2n - 2$, including at most $n - 1$ goods that are temporarily removed by the algorithm (those that are neighbors of removed agents) and at most $n - 1$ goods that remain after the last iteration. *It is crucial that the current partial allocation satisfies envy-freeness* (which can be easily verified for each iteration based on the property of minimal unmatchable groups). This allows us to reduce our problem to the case with at most $2n - 2$ indivisible goods, as we can fix the allocation of the indivisible goods allocated by the above procedure. In Appendix B.2, we formally describe this procedure and the reduction to $m \leq 2n - 2$.

The most technical part comes to the handling of the case with $m \leq 2n - 2$, which is described in Appendix B.3. The case with $m \leq n$ has already been handled in Section 4.3. The case where $n < m \leq 2n - 2$ is handled with a similar idea of comparing to a baseline allocation but requiring a substantial amount of additional effort. We again find a minimal unmatchable group Z of agents, and agents in Z and in $U \setminus Z$ are handled separately in the “last two rounds”. Sample a permutation π uniformly at random. In the first round, agents with “higher priority” in Z receive “the last large good”, while all agents in $U \setminus Z$ receive large goods. In the second round, we allocate agents with “higher priority” one good in the Round-Robin way. Finally, the water-filling process is applied to allocate the divisible good. The analysis for ex-ante proportionality is the most tricky part. Different baseline allocations are used for agents in Z and agents in $U \setminus Z$.

5 BoBW Fairness with Indivisible Goods: Ex-Ante EF + Ex-Post EFX + Ex-Post fPO

In this section, we focus on the indivisible-goods setting and investigate the best-of-both-world guarantee for both fairness and efficiency when the agents have bi-valued utilities. Here, we will assume $a > 0$ for a clearer demonstration. For the case of $a = 0$, it can be reduced to the binary setting and has been addressed [Babaioff et al., 2021]. Our main result is the following:

Theorem 5.1. *In the indivisible-goods setting where agents have bi-valued utilities, Algorithm 2 computes in polynomial time an integral allocation sampled from a randomized allocation that is ex-ante EF, ex-post EFX and ex-post fPO.*

Theorem 5 of Aziz et al. [2024a] presented an instance with two goods and two agents with additive utilities to show that no randomized allocation is simultaneously ex-ante EF, ex-post EF1, and ex-post fPO. Their impossibility result continues to hold when replacing ex-ante EF by ex-ante PROP. It is worth noting that the instance consists of three possible values for the goods, which can be considered in the settings allowing tri-valued utilities or *personalized* bi-valued utilities. Their impossibility result indicates that the assumption of bi-valued utilities is *necessary* in order to achieve best-of-both-world fairness *and* economic efficiency. For tri-valued or personalized bi-valued utilities, Aziz et al.’s impossibility result holds for the weaker notions of ex-ante PROP and ex-post EF1, with two agents and two goods. For bi-valued utilities, our Theorem 5.1 indicates

that the best-of-both-worlds fairness can hold for the stronger notions of ex-ante EF and ex-post EFX, with an arbitrary number of agents and goods. Our result shows a sharp contrast between bi-valued and tri-valued utilities, and thereby completes the whole picture.

5.1 Technique Discussion and Detailed Descriptions of Algorithm 2

For bi-valued utilities, Garg and Murhekar [2023] devised a polynomial-time algorithm to find an EFX and fPO allocation. Their algorithm updates the allocation through consecutive transfers of goods along a path, a process seemingly challenging to implement for achieving an ex-ante fairness guarantee. We instead draw inspiration from the MATCH&FREEZE algorithm of Amanatidis et al. [2021], which computes an EFX allocation for bi-valued utilities in polynomial time. It is worth noting that the MATCH&FREEZE need not return a PO allocation [Garg and Murhekar, 2023, Appendix A.2]. In what follows, we present the first polynomial-time algorithm which can return an ex-ante EF, ex-post EFX, and ex-post fPO allocation when agents have bi-valued utilities. When describing our Algorithm 2, we use the term “round” interchangeably with each execution of the loop in line 4.

We first introduce three possible states for the agents: *active*, *quiet* and *frozen*. Initially, all agents are *active*. When executing lines 13 to 19, if an agent $i \in Z_t$ is allocated a large good in line 14, she will become *frozen* for the next $\lfloor b/a - 1 \rfloor$ rounds, indicating that she will not receive any good in these $\lfloor b/a - 1 \rfloor$ rounds. After those rounds, she will become *quiet*. If $i \in Z_t$ does not receive a large good at this round, she will directly become *quiet* in line 17. Intuitively, during the execution of the loop in lines 4 to 21, an agent i can only accept small goods after she turns quiet, and the count for these goods is recorded in c_i . We also refer to active and quiet agents as *unfrozen* agents.

We now describe Algorithm 2. Lines 1-3 initialize the algorithm, where c_i records the number of reserved small goods for agent i , crucial for the market clear in line 26. The counter cnt records the total number of reserved goods, precisely the sum of all c_i 's during the loop (lines 4-21). A permutation π is generated uniformly at random for subsequent allocations.

The main algorithm proceeds through a multi-round procedure in lines 4-21. In each round, we first check whether the remaining goods in M are sufficient to match one to each unfrozen agent in line 4. If this condition is met, we attempt to find a specific minimal unmatchable group Z_t such that all agents in Z_t have value a over each good matched to agents in $N \setminus Z_t$ in lines 5-9.⁷ This is to ensure the efficiency guarantee by allocating more large goods intuitively.

Given Z_t , if $|Z_t| = 0$, indicating that there exists a perfect matching between all agents in N and the current set of goods in M , lines 10-11 are executed to add this matching directly. When Z_t is non-empty, we first find a maximum matching between Z_t and $\Gamma(Z_t)$ using the augmenting path technique in the reversed order of π (lines 13-17). All goods in $\Gamma(Z_t)$ can be allocated to some agent from the construction of Z_t (see Algorithm 4). The reversed order in line 13 ensures ex-post EFX after the final allocation step in lines 22-25. We then find a perfect matching between $N \setminus Z_t$ and $M \setminus \Gamma(Z_t)$ in lines 18-19 and match these goods accordingly. At the end of each round, we maintain c_i for each quiet agent i and the counter cnt , to allocate a good to those who have not received a good at this round *virtually* (lines 20-21).

If there are not enough goods to allocate one to each unfrozen agent, the while-loop in line 4 is terminated and the final allocation step in lines 22-25 is executed. At this step, a large good is allocated to the first $|M| - \text{cnt}$ unfrozen agents under π as far as possible. Line 22 can adjust the previous allocation for the remaining agents in N for a better efficiency guarantee. Line 25 updates the number of reserved small goods for an agent if she cannot be matched to any large

⁷The relevant concepts can be found in Appendix B.1.

good. Line 26 clears the market based on the reserved number c_i for each agent i , which allocates these goods *actually*. We can then obtain the desired randomized allocation.

Throughout the algorithm, the randomization of our algorithm (the choice of π) only influences the allocation in lines 13-19 and 22-25, which would not affect the assignment of agents to groups Z_t (resp., goods to $\Gamma(Z_t)$). This observation will be helpful for the analysis of the ex-ante property.

5.2 Fairness

Before presenting the results, we first provide some properties of the allocation returned by Algorithm 2. Let N^t and M^t be the corresponding set N and M at the beginning of round t .

Observation 5.2. For each agent $i \in N$,

- she receives exactly one good at each unfrozen round;
- if she belongs to some Z_t which is removed from N in line 19 (assuming this corresponds to round r_i), she values each good allocated to her before round r_i at b and each good in A_j for $j \in N^{r_i} \setminus Z_t$ and $M^{r_i} \setminus \Gamma(Z_t)$ at a ;
- if she does not belong to any Z_t , she values each good allocated to her before the termination of the while-loop in line 4 at b .

According to these observations, we first show that each output integral allocation by our algorithm is EFX. To prove this, we adopt the induction to maintain the following property after the matching at each round: for each agent i , this agent will not envy other agents j except for the case that agents i and j are in the same Z_t and agent i has never been frozen while agent j has been frozen. Under this case, agent i does not envy j using the EFX criteria. Together with the analysis for the final allocation step, we conclude the lemma.

Lemma 5.3. *Every integral allocation returned by Algorithm 2 is EFX.*

We then come to the ex-ante fairness guarantee of our algorithm. The idea is to prove any envy from an agent i to another agent j under π can be eliminated by the gain under another permutation π' which just exchanges i and j in π .

Lemma 5.4. *Algorithm 2 returns an ex-ante EF allocation.*

5.3 Efficiency

We leverage the Fisher market (relevant concepts in Appendix) to provide our efficiency guarantee.

Lemma 5.5. *Every integral allocation returned by Algorithm 2 is fPO.*

Proof. To leverage Theorem C.1 for proving this, our goal is to find a proper price vector \mathbf{p} for each realized integral allocation \mathcal{A} such that $(\mathcal{A}, \mathbf{p})$ is a market equilibrium of a Fisher market. Here, we can directly set the budget e_i for each agent $i \in N$ as $\sum_{g \in A_i} p_g$ to ensure the market clear. It suffices to find \mathbf{p} and show that each agent $i \in N$ only receives goods in MBB_i .

Denote the set of agents not in any previous Z_t by N^r . We assume $m \geq n$ here and the analysis for the corner case when $m < n$ is deferred to Appendix. Define the price vector \mathbf{p} as follows:

- for each agent i in some Z_t , we set $p_g = u_i(g)$ for $g \in A_i$;

- for each agent $i \in N^r$ who receives both large and small goods, set $p_g = u_i(g)$ for each $g \in A_i$.

For the prices of the remaining goods, we will set them iteratively: if there exists one agent i without pricing her goods such that some agent who has priced her goods before values some good in A_i at b , we set $p_g = u_i(g)$ for each $g \in A_i$. If there is no further pricing that can be made, we set the price as a for each remaining good and we call each agent who owns these goods *low-price agent*. We then need to show each agent $i \in N$ only receives goods in MBB_i .

For each agent i in some Z_t , the ratio $u_i(g)/p_g$ for each good $g \in A_i$ is exactly 1. For each good g which is allocated to some agent j in a $Z_{t' \leq t}$, if $p_g = u_j(g) = a$, this good is *actually* allocated at line 26 and g is in $M^i \setminus \Gamma(Z_t)$, which is valued a by i from the second term in Observation 5.2, whose ratio $u_i(g)/p_g$ is at most 1. For each good g which is allocated to some agent j in $N^r \setminus Z_t$, we also have $u_i(g) = a$ from Observation 5.2. Thus, each agent i in some Z_t only receives goods in MBB_i .

For each low-price agent, from the third term in Observation 5.2, all goods allocated to her in the multi-round procedure must be large. Because this agent is not priced before the end of the iteration, she receives no small goods. Since the goods she owns are priced at a which is the lowest price we set and all goods in her bundle must be valued at b under her valuation, all goods in her bundle must be in MBB_i .

We then come to some agent $i \in N^r$ who sets $p_g = u_i(g)$ for each $g \in A_i$. The ratio $u_i(g)/p_g$ for each good $g \in A_i$ is exactly 1. If she values a good g in some $A_j, j \in N$ at b (thus j cannot be a low-price agent otherwise it will be priced before the end from the iteration) and $u_j(g) = p_g = a$, we can backward the iteration from agent i and finally achieve an agent $i' \in N^r$ who receives both large and small goods from her perspective. This backward path along with the good presented in the condition of the iteration is a feasible augmenting path at line 24 and agent i' would not receive a small good, which leads to a contradiction. This completes the proof for the case when $m \geq n$. \square

We are now ready to prove our main result in this section.

Proof of Theorem 5.1. From Lemmas 5.3, 5.4 and 5.5, it suffices to show Algorithm 2 can terminate in polynomial time. From the description of our algorithm, since $|M|$ keeps decreasing which bounds the total number of rounds by m and the only thing we need is to show lines 6-9 can be terminated in polynomial time. From the construction of Z_t according to Algorithm 4, if we keep the perfect matching between $N \setminus Z_t$ and $M \setminus \Gamma(Z_t)$ by just replacing the good g' by g , one edge from $j \in Z_t$ to g' is added, this leads to one of the following cases: either (1) one augmenting path can be performed and the size of the maximum matching between N and M is increased by 1, or (2) there is no additional augmenting path but $|\Gamma(Z_t)|$ is increased by at least 1. After at most n times of the case (2) occurs, case (1) will occur. From the total number of the occurrences of (1) is at most n , lines 6-9 can be terminated in $O(n^2)$ loops. \square

6 Conclusion

In this paper, we have studied the best-of-both-worlds fairness for indivisible-goods and mixed-goods allocations. With indivisible goods, we provided polynomial-time algorithms that achieve ex-ante EF and ex-post EFX allocation for two agents and ex-ante EF, ex-post EFX, and ex-post fPO for n agents with bi-valued utilities. With mixed goods, we showed polynomial-time algorithms

that achieve ex-ante EF and ex-post EFM allocation for two agents and ex-ante PROP and ex-post EFM for n agents with bi-valued utilities.

In future research, it would be interesting to further strengthen the results in this paper, the most intriguing of which is perhaps the (in)compatibility between ex-ante EF and ex-post EFM in the mixed-goods setting. For mixed goods, another interesting direction is to extend BoBW-fair study to maximin share (MMS) guarantee [Bei et al., 2021b].

Acknowledgements

The authors would like to thank Jiaxin Song for his great suggestions on this paper.

This work was partially supported by ARC Laureate Project FL200100204 on “Trustworthy AI”, by the National Natural Science Foundation of China (Grant No. 62102117), by the Shenzhen Science and Technology Program (Grant No. GXWD20231129111306002), by the Guangdong Basic and Applied Basic Research Foundation (Grant No. 2023A1515011188), and by the National Natural Science Foundation of China (No. 62102252).

References

- Atila Abdulkadiroğlu and Tayfun Sönmez. Random serial dictatorship and the core from random endowments in house allocation problems. *Econometrica*, 66(3):689–701, 1998.
- Mohammad Akbarpour and Afshin Nikzad. Approximate random allocation mechanisms. *The Review of Economic Studies*, 87(6):2473–2510, 2020.
- Hannaneh Akrami, Noga Alon, Bhaskar Ray Chaudhury, Jugal Garg, Kurt Mehlhorn, and Ruta Mehta. EFX: A simpler approach and an (almost) optimal guarantee via rainbow cycle number. In *Proceedings of the 24th ACM Conference on Economics and Computation (EC)*, page 61, 2023a.
- Hannaneh Akrami, Kurt Mehlhorn, Masoud Seddighin, and Golnoosh Shahkarami. Randomized and deterministic maximin-share approximations for fractionally subadditive valuations. In *Proceedings of the 37th Annual Conference on Neural Information Processing Systems (NeurIPS)*, pages 58821–58832, 2023b.
- Hannaneh Akrami, Jugal Garg, Eklavya Sharma, and Setareh Taki. Improving approximation guarantees for maximin share. In *Proceedings of the 25th ACM Conference on Economics and Computation (EC)*, 2024. Forthcoming.
- Georgios Amanatidis, Evangelos Markakis, and Apostolos Ntokos. Multiple birds with one stone: Beating $1/2$ for EFX and GMMS via envy cycle elimination. *Theoretical Computer Science*, 841: 94–109, 2020.
- Georgios Amanatidis, Georgios Birmpas, Aris Filos-Ratsikas, Alexandros Hollender, and Alexandros A. Voudouris. Maximum Nash welfare and other stories about EFX. *Theoretical Computer Science*, 863:69–85, 2021.
- Georgios Amanatidis, Haris Aziz, Georgios Birmpas, Aris Filos-Ratsikas, Bo Li, Hervé Moulin, Alexandros A. Voudouris, and Xiaowei Wu. Fair division of indivisible goods: Recent progress and open questions. *Artificial Intelligence*, 322:103965, 2023.

- Haris Aziz, Aditya Ganguly, and Evi Micha. Best of both worlds fairness under entitlements. In *Proceedings of the 22nd International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 941–948, 2023a.
- Haris Aziz, Xinhang Lu, Mashbat Suzuki, Jeremy Vollen, and Toby Walsh. Best-of-both-worlds fairness in committee voting. In *Proceedings of the 19th Conference on Web and Internet Economics (WINE)*, page 676, 2023b.
- Haris Aziz, Rupert Freeman, Nisarg Shah, and Rohit Vaish. Best of both worlds: Ex-ante and ex-post fairness in resource allocation. *Operations Research*, 72(4):1674–1688, 2024a.
- Haris Aziz, Xinhang Lu, Mashbat Suzuki, Jeremy Vollen, and Toby Walsh. Fair lotteries for participatory budgeting. In *Proceedings of the 38th AAAI Conference on Artificial Intelligence (AAAI)*, pages 9469–9476, 2024b.
- Moshe Babaioff, Tomer Ezra, and Uriel Feige. Fair and truthful mechanisms for dichotomous valuations. In *Proceedings of the 35th AAAI Conference on Artificial Intelligence (AAAI)*, pages 5119–5126, 2021.
- Moshe Babaioff, Tomer Ezra, and Uriel Feige. On best-of-both-worlds fair-share allocations. In *Proceedings of the 18th Conference on Web and Internet Economics (WINE)*, pages 237–255, 2022.
- Xiaohui Bei, Zihao Li, Jinyan Liu, Shengxin Liu, and Xinhang Lu. Fair division of mixed divisible and indivisible goods. *Artificial Intelligence*, 293:103436, 2021a.
- Xiaohui Bei, Shengxin Liu, Xinhang Lu, and Hongao Wang. Maximin fairness with mixed divisible and indivisible goods. *Autonomous Agents and Multi-Agent Systems*, 35(2):34, 2021b.
- Ben Berger, Avi Cohen, Michal Feldman, and Amos Fiat. Almost full EFX exists for four agents. In *Proceedings of the 36th AAAI Conference on Artificial Intelligence (AAAI)*, pages 4826–4833, 2022.
- Umang Bhaskar, A. R. Sricharan, and Rohit Vaish. On approximate envy-freeness for indivisible chores and mixed resources. In *Proceedings of the 24th International Conference on Approximation Algorithms for Combinatorial Optimization Problems (APPROX)*, pages 1:1–1:23, 2021.
- Anna Bogomolnaia and Hervé Moulin. A new solution to the random assignment problem. *Journal of Economic Theory*, 100(2):295–328, 2001.
- Xiaolin Bu, Jiabin Song, and Ziqi Yu. EFX allocations exist for binary valuations. In *Proceedings of the International Joint Conference on Theoretical Computer Science – Frontier of Algorithmic Wisdom (IJTCS-FAW)*, pages 252–262, 2023.
- Xiaolin Bu, Zihao Li, Shengxin Liu, Xinhang Lu, and Biaoshuai Tao. Best-of-both-worlds fair allocation of indivisible and mixed goods. In *Proceedings of the 20th Conference on Web and Internet Economics (WINE)*, 2024. Forthcoming.
- Eric Budish. The combinatorial assignment problem: Approximate competitive equilibrium from equal incomes. *Journal of Political Economy*, 119(6):1061–1103, 2011.
- Eric Budish, Yeon-Koo Che, Fuhito Kojima, and Paul Milgrom. Designing random allocation mechanisms: Theory and applications. *American Economic Review*, 103(2):585–623, 2013.

- Ioannis Caragiannis, Nick Gravin, and Xin Huang. Envy-freeness up to any item with high Nash welfare: The virtue of donating items. In *Proceedings of the 20th ACM Conference on Economics and Computation (EC)*, pages 527–545, 2019a.
- Ioannis Caragiannis, David Kurokawa, Hervé Moulin, Ariel D. Procaccia, Nisarg Shah, and Junxing Wang. The unreasonable fairness of maximum Nash welfare. *ACM Transactions on Economics and Computation*, 7(3):12:1–12:32, 2019b.
- Bhaskar Ray Chaudhury, Telikepalli Kavitha, Kurt Mehlhorn, and Alkmini Sgouritsa. A little charity guarantees almost envy-freeness. *SIAM Journal on Computing*, 50(4):1336–1358, 2021.
- Bhaskar Ray Chaudhury, Jugal Garg, Kurt Mehlhorn, Ruta Mehta, and Pranabendu Misra. Improving envy freeness up to any good guarantees through rainbow cycle number. *Mathematics of Operations Research*, 2023. Forthcoming.
- Bhaskar Ray Chaudhury, Jugal Garg, and Kurt Mehlhorn. EFX exists for three agents. *Journal of the ACM*, 71(1):4:1–4:27, 2024.
- Michal Feldman, Simon Mauras, Vishnu V. Narayan, and Tomasz Ponitka. Breaking the envy cycle: Best-of-both-worlds guarantees for subadditive valuations. In *Proceedings of the 25th ACM Conference on Economics and Computation (EC)*, 2024. Forthcoming. Extended version available at <https://arxiv.org/abs/2304.03706>.
- Duncan Karl Foley. Resource allocation and the public sector. *Yale Economics Essays*, 7(1):45–98, 1967.
- Jugal Garg and Aniket Murhekar. Computing fair and efficient allocations with few utility values. *Theoretical Computer Science*, 962:113932, 2023.
- Jugal Garg and Eklavya Sharma. Best-of-both-worlds fairness of the envy-cycle-elimination algorithm. arXiv:2410.08986, 2024. URL <https://arxiv.org/abs/2410.08986>.
- Paul Gözl, Dominik Peters, and Ariel D. Procaccia. In this apportionment lottery, the house always wins. *Operations Research*, 2024. Forthcoming.
- Daniel Halpern, Ariel D. Procaccia, Alexandros Psomas, and Nisarg Shah. Fair division with binary valuations: One rule to rule them all. In *Proceedings of the 16th International Conference on Web and Internet Economics (WINE)*, pages 370–383, 2020.
- Martin Hoefer, Marco Schmalhofer, and Giovanna Varricchio. Best of both worlds: Agents with entitlements. In *Proceedings of the 22nd International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 564–572, 2023.
- Bo Li, Zihao Li, Shengxin Liu, and Zekai Wu. Allocating mixed goods with customized fairness and indivisibility ratio. In *Proceedings of the 33rd International Joint Conference on Artificial Intelligence (IJCAI)*, pages 2868–2876, 2024a.
- Zihao Li, Shengxin Liu, Xinhang Lu, and Biaoshuai Tao. Truthful fair mechanisms for allocating mixed divisible and indivisible goods. In *Proceedings of the 32nd International Joint Conference on Artificial Intelligence (IJCAI)*, pages 2808–2816, 2023.
- Zihao Li, Shengxin Liu, Xinhang Lu, Biaoshuai Tao, and Yichen Tao. A complete landscape for the price of envy-freeness. In *Proceedings of the 23rd International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 1183–1191, 2024b.

- Richard J. Lipton, Evangelos Markakis, Elchanan Mossel, and Amin Saberi. On approximately fair allocations of indivisible goods. In *Proceedings of the 5th ACM Conference on Electronic Commerce (EC)*, pages 125–131, 2004.
- Shengxin Liu, Xinhang Lu, Mashbat Suzuki, and Toby Walsh. Mixed fair division: A survey. *Journal of Artificial Intelligence Research*, 80:1373–1406, 2024.
- Ryoga Mahara. Existence of EFX for two additive valuations. *Discrete Applied Mathematics*, 340:115–122, 2023.
- Andreu Mas-Colell, Michael Dennis Whinston, Jerry R Green, et al. *Microeconomic theory*. Oxford University Press, 1995.
- Koichi Nishimura and Hanna Sumita. Envy-freeness and maximum Nash welfare for mixed divisible and indivisible goods. arXiv:2302.13342, 2023. URL <https://arxiv.org/abs/2302.13342>.
- Benjamin Plaut and Tim Roughgarden. Almost envy-freeness with general valuations. *SIAM Journal on Discrete Mathematics*, 34(2):1039–1068, 2020.
- Hugo Steinhaus. The problem of fair division. *Econometrica*, 16(1):101–104, 1948.
- Warut Suksompong. Constraints in fair division. *ACM SIGecom Exchanges*, 19(2):46–61, 2021.
- Warut Suksompong. Weighted fair division of indivisible items: A review. *Information Processing Letters*, 187:106519, 2025.
- Mashbat Suzuki and Jeremy Vollen. Maximum flow is fair: A network flow approach to committee voting. In *Proceedings of the 25th ACM Conference on Economics and Computation (EC)*, 2024. Forthcoming.
- Hal R. Varian. Equity, envy, and efficiency. *Journal of Economic Theory*, 9(1):63–91, 1974.

Algorithm 2: Bi-valued indivisible goods: Ex-ante EF, ex-post EFX, and ex-post fPO

Input: Agents N , indivisible goods M , and agents' utility functions.

```
1  $\forall i \in [n], A_i \leftarrow \emptyset, c_i \leftarrow 0$ 
2  $t \leftarrow 1, \text{cnt} \leftarrow 0$ 
3 Uniformly generate a random permutation  $\pi$  of the agents.
4 while  $|M| \geq [\# \text{ unfrozen agents}] + \text{cnt}$  do
5   Find the minimal unmatchable group  $Z_t$  among  $N$  and  $M$  according to Algorithm 4.
6   while  $|Z_t| > 0$  and there exists  $i \in N \setminus Z_t$  and  $g \in A_i$  s.t.  $u_j(g) = b$  for an agent  $j \in Z_t$  do
7     Let  $g' \in M$  be the good matched to agent  $i$  in a perfect matching between  $N \setminus Z_t$ 
8     and  $M \setminus \Gamma(Z_t)$ .
9      $M \leftarrow M \cup \{g\} \setminus \{g'\}, A_i \leftarrow A_i \cup \{g'\} \setminus \{g\}$ 
9     Find the minimal unmatchable group  $Z_t$  among  $N$  and  $M$  via Algorithm 4.
10  if  $|Z_t| = 0$  then
11    Find a perfect matching between  $N$  and  $M$ . For each agent  $i \in N$ , add her matched
11    good to  $A_i$ . Remove the matched goods from  $M$ .
12  else
13    foreach  $i \in Z_t$  in the reversed order of  $\pi$  do
14      if  $i$  can find an augmenting path to a good in  $\Gamma(Z_t)$  then
15        Update  $\mathcal{A}^\pi$  according to this path.
16        Freeze  $i$  for the next  $\lfloor b/a - 1 \rfloor$  rounds.
17      else Set  $i$  as quiet.
18    Match the corresponding good to each agent in  $N \setminus Z_t$  in a perfect matching
18    between  $N \setminus Z_t$  and  $M \setminus \Gamma(Z_t)$ .
19     $N \leftarrow N \setminus Z_t, M \leftarrow M \setminus \Gamma(Z_t), t \leftarrow t + 1$ 
20     $\text{cnt} \leftarrow \text{cnt} + [\# \text{ quiet agents}]$ 
21     $c_i \leftarrow c_i + 1$  for each quiet agent  $i$ .
22 Let  $M' \leftarrow M \cup \bigcup_{i \in N} A_i$  and match each  $g \in A_i$  for some  $i \in N$  to a copy of agent  $i$  initially.
23 foreach agent  $i$  in the first  $|M| - \text{cnt}$  of the order  $\pi$  after removing the frozen agents do
24   if  $i$  can find an augmenting path to a good in  $M'$  then Update  $\mathcal{A}^\pi$  according to this
24   augmenting path.
25   else  $c_i \leftarrow c_i + 1$ 
26 Allocate the remaining goods arbitrarily based on  $c_i$ .
27 return Allocation  $\mathcal{A}^\pi$ 
```

Algorithm 3: An ex-ante EF and ex-post EFX randomized allocation for two agents

Input: Agents $N = [2]$ and indivisible goods M

- 1 For each $i \in [2]$, $\mathcal{A}^i \leftarrow \text{LocalSearch}(\emptyset, M, u_i)$
- 2 **if** $\exists i \in [2]$ such that $u_i(\mathcal{A}_1^i) = u_i(\mathcal{A}_2^i)$ or $u_{3-i}(\mathcal{A}_1^i) \geq u_{3-i}(\mathcal{A}_2^i)$ **then**
- 3 **return** $\{(1, \mathcal{A}^i)\}$ where agent $3 - i$ picks her preferred bundle first
- 4 **while** $\exists i \in [2]$ such that $u_{3-i}(\mathcal{A}_2^i) - u_{3-i}(\mathcal{A}_1^i) < u_{3-i}(\mathcal{A}_2^{3-i}) - u_{3-i}(\mathcal{A}_1^{3-i})$ **do**
- 5 $\mathcal{A}^{3-i} \leftarrow \text{LocalSearch}(\mathcal{A}_1^i, \mathcal{A}_2^i, u_{3-i})$
- 6 **if** $\exists j \in [2]$ such that $u_j(\mathcal{A}_1^j) = u_j(\mathcal{A}_2^j)$ or $u_{3-j}(\mathcal{A}_1^j) \geq u_{3-j}(\mathcal{A}_2^j)$ **then**
- 7 **return** $\{(1, \mathcal{A}^j)\}$ where agent $3 - j$ picks her preferred bundle first
- 8 **if** \mathcal{A}^{3-i} is also EFX under u_i **then**
- 9 $\mathcal{A}^i \leftarrow \mathcal{A}^{3-i}$
- 10 **return** $\{(0.5, \mathcal{A}^1), (0.5, \mathcal{A}^2)\}$ where agent $3 - i$ picks her preferred bundle first in the realized allocation \mathcal{A}^i for each $i \in [2]$
- 11 **return** $\{(0.5, \mathcal{A}^1), (0.5, \mathcal{A}^2)\}$ where agent $3 - i$ picks her preferred bundle first in the realized allocation \mathcal{A}^i for each $i \in [2]$

12 **Function** $\text{LocalSearch}(A, B, u)$:

- 13 **if** $u(A) > u(B)$ **then** Swap A and B .
- 14 **while** $\exists g \in B$ such that $u(A \cup \{g\}) < u(B)$ **do**
- 15 $g \leftarrow \arg \max_{g' \in B \text{ and } u(A \cup \{g'\}) < u(B)} u(g')$
- 16 $A \leftarrow A \cup \{g\}, B \leftarrow B \setminus \{g\}$
- 17 **if** $u(A) > u(B)$ **then** Swap A and B .
- 18 **return** (A, B)

A Omitted Proofs in Section 3

A.1 Proof of Lemma 3.1

The termination condition ensures that in the output allocation (A', B') , we have $u(A') \leq u(B')$ and for every good $g \in B'$ we have $u(A' \cup \{g\}) \geq u(B')$, which meets the EFX condition. It suffices to show this subroutine can terminate in polynomial time.

Without loss of generality, we assume $u(A) \leq u(B)$ in the initial allocation (A, B) . When executing line 16, we choose the item $g \in B$ with the highest value according to u such that $u(A \cup \{g\}) < u(B)$ and move it from B to A . Before the next swap step, both the utility difference between the two bundles and the size of B are decreasing. If the swap step (line 17) is never executed, the number of steps is upper bounded by m . If one swap step (line 17) is executed, assume the last item moved from B to A is g . This indicates that, before moving the item g from bundle B to bundle A , we have $u(B) - u(A) > u(g)$ and $u(A) + u(g) > u(B) - u(g)$ from the conditions in lines 14 and 17, so the utility difference between the two bundles decreases from a value larger than $u(g)$ to a value less than $u(g)$ after moving the item g . Thus, during the whole subroutine, the utility difference keeps decreasing, and an item g' will not be further moved between the two bundles if $u(g') \geq u(g)$. The values of the items that have been moved are at least $u(g)$ (due to line 15), so an item can be moved between the two bundles at most once. The overall time complexity is bounded by $O(m \log m)$, where we may first sort the items according to

u , and then perform the above operations in linear time.

A.2 Proof of Theorem 3.2

If the allocation is returned in line 10, this is ex-post EFX from Lemma 3.1 and the condition in line 8, and trivially ex-ante EF since we only exchange two bundles in these two realizations. If the allocation is returned in lines 3 and 7, either the agent i in line 2 or the agent j in line 6 treats two bundles equally or the other agent will choose the bundle which is less valued by this agent. In both cases, this is exactly EF, in both ex-ante and ex-post senses. If the allocation is returned in line 11 (which also means the conditions in lines 2 and 6 fail), from Lemma 3.1, both \mathcal{A}^1 and \mathcal{A}^2 are EFX if for each $i \in [2]$, agent $3 - i$ picks her preferred bundle first in \mathcal{A}^i . Because condition in line 4 is violated, for each agent $i \in [2]$, we have $u_i(A_2^i) - u_i(A_1^i) \leq u_i(A_2^{3-i}) - u_i(A_1^{3-i})$, so the loss of $u_i(A_2^i) - u_i(A_1^i)$ for agent i under \mathcal{A}^i is upper bounded by the gain of $u_i(A_2^{3-i}) - u_i(A_1^{3-i})$ under \mathcal{A}^{3-i} . Thus, this randomized allocation is ex-ante EF and ex-post EFX. It suffices to show that this algorithm can terminate in polynomial steps, where we adopt a similar analysis.

From the condition at line 4 and the monotonic property in the proof of Lemma 3.1, after executing line 5, the utility difference between the two bundles in \mathcal{A}^{3-i} under u_{3-i} decreases (to see this, Lemma 3.1 implies that the utility difference between the two bundles in the updated \mathcal{A}^{3-i} is weakly smaller than the utility difference of the two bundles in \mathcal{A}^i under u_{3-i} , and the condition at line 4 implies that the utility difference between the two bundles in \mathcal{A}^{3-i} strictly decreases after executing line 5). We first notice that line 13 cannot be executed in this call of the subroutine because of the failure of the conditions at line 2 or the previously executed line 6. If there is no swap step (line 17) executed in this call of the subroutine, since the original allocation (A_1^i, A_2^i) is EFX under u_i , either $u_i(A_1^{3-i}) \geq u_i(A_2^{3-i})$ and then meets the condition at line 6, or the allocation \mathcal{A}^{3-i} is also EFX for agent i . In both cases, the algorithm terminates. Thus, we then assume some swap step is executed in this call of subroutine, which means that when the item last transferred from one bundle to another under u_{3-i} is g , the difference between the utilities of two bundles in \mathcal{A}^{3-i} under u_{3-i} is from a value larger than $u_{3-i}(g)$ to a value less than $u_{3-i}(g)$.

The difference for each $\mathcal{A}^i, i \in [2]$ under u_i is decreasing during the whole process before terminating. Similar to the analysis in Appendix A.1, each item that has been moved between the two bundles will not be further moved by the same agent. Thus, the total number of move operations is bounded by $2m$. The overall time complexity is $O(m \log m)$, where we may first sort the items according to u_1 and u_2 respectively in $O(m \log m)$, and then perform the above operations in linear time. For the execution of line 8, we can use a heap to maintain the item with the smallest utility in $O(\log m)$ time per operation. Since the total number of executions of this line can also be bounded by the total number of move operations, which is $O(m)$, the overall time complexity for this line remains $O(m \log m)$.

A.3 Proof of Theorem 3.3

We analyze the procedure described before the statement of this theorem. In the first case where a single integral allocation is given (lines 3 and 7), this is an EF allocation and thus ex-ante EF and ex-post EFM. We then assume there exist two allocations \mathcal{A}^1 and \mathcal{A}^2 where \mathcal{A}^i is EFX under u_i for each $i \in [2]$. If there exists an agent $i \in [2]$ such that the good d is in bundle A_2^i , from the ex-post EFX property, we have $u_i(A_2^i) \geq u_i(A_1^i)$ and $u_i(A_2^i) - u_i(d) \leq u_i(A_1^i)$, thus there exists an $\alpha \in [0, 1]$ such that agent i treats these two bundles equally after transferring an α -fraction of the divisible good d . Thus, let agent $3 - i$ pick her preferred bundle first and we achieve an EF allocation.

Algorithm 4: Finding minimal unmatchable group

Input: Agents N , indivisible goods M , and agents' utility functions.

- 1 Construct a bipartite graph $G = (N, M, E)$ where $(i, g) \in E$ if $u_i(g) = b$.
 - 2 Find a maximum matching on G .
 - 3 Let Z be the set of all reachable agents from all unmatched agents through alternating paths.
 - 4 **return** *The minimal unmatchable group Z*
-

For the last case where the good d is in the bundle with smaller utility for both agents, since each integral allocation in the support is EFX and there is no divisible good in the better bundle, this can also imply EFM.

B Omitted Details in Section 4

In this section, we provide the remaining details for the proof of Theorem 4.1.

B.1 Finding the Minimal Unmatchable Group

In this section, we define the notion of *minimal unmatchable group* of agents and describe an algorithm for finding such a group. These will be used in the next two sections.

A bipartite graph $G = (U, V, E)$ is constructed, where U denotes the set of agents, V denotes the set of indivisible goods, and $(i, g) \in E$ if $u_i(g) = b$. We use $\Gamma(S)$ for $S \subseteq U$ to denote the set of neighbors of S in G . Given a bipartite graph and a matching, an alternating path and an augmenting path are defined as follows.

Definition B.1 (Alternating path in a bipartite graph). Given a matching in a bipartite graph, an *alternating path* is a path that begins with an unmatched agent or an unmatched good and in which the edges belong alternately to the matching and not to the matching.

Definition B.2 (Augmenting path in a bipartite graph). An *augmenting path* is an alternating path starting from an unmatched agent (resp., good) and terminating at an unmatched good (resp., agent).

Definition B.3 (Perfect matching in a bipartite graph). Given a bipartite graph G , a *perfect matching* with respect to a set of agents $U' \subseteq U$ and a set of goods $V' \subseteq V$ is a matching that matches each agent in U' to a unique good in V' .

Based on the bipartite graph G , we define the *minimal unmatchable group* in Definition B.4.

Definition B.4 (Minimal unmatchable group). A group of agents Z is said to be a *minimal unmatchable group* if $G = (U, V, E)$ contains a perfect matching between $U \setminus Z$ and $V \setminus \Gamma(Z)$ and there is no subset $S \subseteq Z$ such that

- S has a perfect matching in the bipartite graph, and
- there is no edge between each agent $i \in Z \setminus S$ and each good matched to S .

Intuitively, for each agent belonging to the minimal unmatchable group Z , her value to each item except $\Gamma(Z)$ is a . If there exists a perfect matching for all agents, $Z = \emptyset$.

Algorithm 5: Partially allocating indivisible goods

Input: Agents N , indivisible goods M , and agents' utility functions.

```
1  $\forall i \in [n], A'_i \leftarrow \emptyset$ 
2 Let  $T \leftarrow \emptyset$  be the set of agents that do not have a perfect matching.
3 Let  $Y \leftarrow \emptyset$  be  $\Gamma(T)$  among the unallocated goods.
4  $\text{cnt} \leftarrow 0, t \leftarrow 1$ 
5 while  $|M| \geq \text{cnt} + n$  do
6   Find the minimal unmatchable group  $Z_t$  among  $N \setminus T$  and  $M$  according to
   Algorithm 4.
7    $T \leftarrow T \cup Z_t, Y \leftarrow Y \cup \Gamma(Z_t)$ 
8    $M \leftarrow M \setminus \Gamma(Z_t)$  and update the bipartite graph.
9   if  $|M| < \text{cnt} + n$  then break
10  Find a perfect matching between  $N \setminus T$  and  $M$ , and assume each agent  $i \in N \setminus T$  is
   matched to  $g_{i'} \in M$ .
11   $\forall i \in N \setminus T, A_i \leftarrow A_i \cup \{g_{i'}\}$ 
12  Remove the matched items from  $M$ .
13  Update the bipartite graph.
14   $\text{cnt} \leftarrow \text{cnt} + |T|, t \leftarrow t + 1$ 
15  $t \leftarrow t - 1$ 
16 foreach  $i \in [n]$  do
17   Arbitrarily allocate  $t - |A_i|$  goods from  $M$  to  $A_i$ , and remove the allocated goods
   from  $M$ .
18 Let  $V \leftarrow M$ 
19 return Partial allocation  $\mathcal{A}'$ , and unallocated indivisible goods  $Y \cup V$ 
```

A straightforward algorithm to find the minimal unmatchable group is to iteratively add the minimal set of agents that violates Hall's condition into the group. We also provide an alternative approach in Algorithm 4.

B.2 Reduction to $m \leq 2n - 2$

Our first step to handle the general case is to allocate the indivisible goods so that there are at most $2n - 2$ indivisible goods that remain unallocated after this step, and the partial integral allocation satisfies envy-freeness. The pseudocode is shown in Algorithm 5.

In each iteration of the algorithm, we run Algorithm 4 to find the minimal unmatchable group Z . If $Z = \emptyset$, we directly find a perfect matching for all agents and allocate the goods accordingly. Otherwise, Z and $\Gamma(Z)$ (i.e., the neighbors of Z) are removed from the bipartite graph. We find a perfect matching between the remaining agents and the remaining goods, which, by the definition of the minimal unmatchable group, is guaranteed to exist. During the process, a variable cnt is maintained to indicate how many additional indivisible goods should be allocated after each iteration to ensure that each bundle contains the same number of indivisible goods. The termination condition of the `while`-loop in lines 5 and 9 ensures that the goods will not run out. After the number of goods is not enough to guarantee the same size of each bundle for the next iteration, we terminate the process and obtain a partial allocation.

In the following part, we use $\mathcal{A}' = (A'_1, \dots, A'_n)$ to denote this partial allocation, T to denote the set of all the unmatchable agents, and Y to denote the neighbors of T that are removed from

M during the algorithm.

Lemma B.5. $|Y \cup V| \leq 2n - 2$.

Proof. For each minimal unmatchable group Z , we have $Z > \Gamma(Z)$; otherwise, Hall's condition is satisfied, and there is a perfect matching between Z and $\Gamma(Z)$. Hence, $|Y| < |T| \leq n$. Moreover, the number of goods allocated in line 16 is equal to cnt . Hence, the number of goods in V is less than n guaranteed by the termination condition of the `while`-loop, as desired. \square

Lemma B.6. *Algorithm 5 returns a partial integral allocation that is envy-free.*

Proof. First, Algorithm 5 ensures that each agent receives the same number of goods. For each agent in $N \setminus T$, envy-freeness is guaranteed as she only receives goods of value b . For each agent $i \in T$, assume she is added to T at round t_i , then each good she receives before round t_i has value b to her, and each good allocated to any agent after round t_i (including t_i) has value a to her. Hence, $u_i(A_i) = (t_i - 1) \cdot b + (t - t_i + 1) \cdot a$ while $u_i(A_j) \leq (t_i - 1) \cdot b + (t - t_i + 1) \cdot a$, meaning that agent i is envy-free in the partial allocation. \square

We have already shown that the partial integral allocation \mathcal{A}' satisfies EF and thus PROP over goods $M' = \bigcup_{i=1}^n A'_i$. We observe that if there is a randomized allocation \mathcal{A} over $(M \setminus M') \cup D$ that satisfies ex-ante PROP and ex-post EFM, then we can construct the allocation over M where each agent i receives bundle A_i deterministically and the corresponding bundle in \mathcal{A}_j with probability p_j which satisfies ex-ante PROP and ex-post EFM. This reduces our problem to the case with at most $2n - 2$ indivisible goods.

In the next section, we assume there are n agents, at most $2n - 2$ indivisible goods, and a single divisible good $\{d\}$.

B.3 Allocating Remaining Goods

As mentioned, we have assumed $m \leq 2n - 2$. The case where $m \leq n$ has already been handled in Section 4.3. Therefore, we assume $n < m \leq 2n - 2$ below.

Step 1: We run Algorithm 4 on N and M to find the minimal unmatchable group. Denote the group we find by T and their neighbors $\Gamma(T)$ by Y . This guarantees the existence of a perfect matching between $N \setminus T$ and $M \setminus Y$. We then construct a bipartite graph $G = (N, M, E)$ for the agents and the indivisible goods where $(i, g) \in E$ if $u_i(g) = b$, and find a random permutation π of the agents. We denote the first k agents in π by $\pi[:k]$.

We adopt a similar idea as in Round-Robin. As $n < m \leq 2n - 2$, each agent of $\pi[:m - n]$ will receive two goods, and the other agents will receive one good. Meanwhile, we execute some extra operations to ensure the allocation satisfies ex-ante PROP. Next, we describe how to allocate the indivisible goods for agents in $N \setminus T$ (resp., T) in Step 2 (resp., Step 3).

Step 2: First, we decide the allocation of indivisible goods for agents in $N \setminus T$. Consider the induced subgraph $G' = (N \setminus T, M \setminus Y, E')$.

First phase: Find a perfect one-to-one matching in G' and allocate the goods accordingly. Note that the matching is fixed regardless of π .

Second phase: Following the order in π , iteratively let each agent of $\pi[:m - n] \cap (N \setminus T)$ receive an unallocated good with the highest value.

Step 3: Next, we decide the allocation of indivisible goods for agents in T . Each agent receives exactly one good in the first phase and the remaining goods are handled in the second phase.

First phase: We handle the agents in T sequentially according to their order in π . Similar to Step 2, let each agent receive a good of value b if there is still such a good available and add this edge to the matching. Otherwise, find an augmenting path and update the goods along the path as well as the matching. If there is no augmenting path, we skip the current agent and let her receive an arbitrary good after finishing executing the aforementioned process to all of the agents in T .

Second phase: After each agent in T receives a good, we further allocate an unallocated indivisible good of M to each of $\pi[: m - n] \cap T$.

Step 4: Execute the water-filling process for divisible goods and obtain the allocation \mathcal{A}^π . The randomized allocation we obtain is $\{(\frac{1}{n!}, \mathcal{A}^\pi)\}$ and is denoted by \mathcal{R} .

In what follows, we show that the randomized allocation \mathcal{R} is ex-ante PROP, and consider the agents in T and in $N \setminus T$ in Lemmas B.8 and B.9 separately and respectively. We start with a lemma which will be used in the proof of Lemma B.8.

Lemma B.7. *The indivisible goods in Y are fully allocated after the first phase in Step 3.*

Proof. Consider the induced subgraph $G^* = (T, Y, E^*)$. Assume there exists a good $g \in Y$ being unallocated after the first phase. We collect all the alternating paths starting from g and denote the set of vertices belonging to T on the paths as T_1 , where $T_1 \neq \emptyset$ since $g \in \Gamma(T)$. We argue that each agent in T_1 is matched in G^* ; otherwise, there is an augmenting path, and the allocation should have been updated during the first phase of Step 3. Every agent in $T \setminus T_1$ values each good matched to agents T_1 at a as we have already found all the alternating paths starting from g . Let $S \subsetneq T_1$ and $S \neq \emptyset$. Then, S is an evidence demonstrating that T is not the minimal unmatchable group (see Definition B.4), a contradiction. \square

Lemma B.8. *The randomized allocation \mathcal{R} is ex-ante PROP for agents in T .*

Proof. We define baseline allocation as in Definition 4.4, and adopt a similar analysis as in Theorem 4.3. For any agent $i \in T$, as there is no edge from T to $M \setminus Y$ and $|Y| < |T| \leq n$, the number of goods valued at b by agent i is at most n . Hence, each bundle B_k^i where $k \in [n]$ in the baseline allocation \mathcal{B}^i contains at most one indivisible good of value b and the number of bundles that contain two indivisible goods with values b and a respectively is maximized according to the property of Round-Robin.

We use $A(\pi, k)$ to denote the bundle allocated to the agent ranked k -th in the permutation π in the allocation output by our algorithm. Consider an arbitrary permutation π where agent i is ranked k -th (i.e., $\pi(k) = i$), we will show that $u_i(A(\pi, k)) \geq u_i(B_k^i)$. Since an agent in $\pi[: m - n]$ receives two goods and otherwise one good in our algorithm, the number of indivisible goods received by agent i is the same as that in the baseline allocation before the water-filling process with whichever index in π . In addition, she will receive a good with value b if B_k^i contains one good. Hence, $u_i(A(\pi, k)) \geq u_i(B_k^i)$ if B_k^i contains only indivisible goods.

If B_k^i contains a fraction of the divisible good D , let $x = u_i(B_k^i)$. Denote the actual output allocation by $\mathcal{A} = (A_1, \dots, A_n)$, and denote the set of bundles that only contain indivisible goods in the actual allocation \mathcal{A} by $\mathcal{S} \subseteq \mathcal{A}$. By Lemma B.7 and according to the first phase of Step 3, each agent in T receives at most one good from Y , so no bundle in the actual allocation contains two indivisible goods both valued at b to agent i . As the number of bundles with the maximum possible value of indivisible goods (which is $b + a$) is maximized in \mathcal{B}^i , we have $\sum_{A_j \in \mathcal{S}} u_i(A_j) \leq$

$\sum_{j=1}^{|\mathcal{S}|} u_i(B_j^i)$, which holds even if we do not consider all the divisible goods in $\bigcup_{j=1}^{|\mathcal{S}|} B_j^i$. Therefore, if $u_i(A_i) = x' < x$ in the actual allocation \mathcal{A} , we will reach the following contradiction

$$\begin{aligned} u_i(M \cup D) &= \sum_{A_j \in \mathcal{S}} u_i(A_j) + \sum_{A_j \notin \mathcal{S}} u_i(A_j) \leq \sum_{A_j \in \mathcal{S}} u_i(A_j) + (n - |\mathcal{S}|) \cdot x' \\ &< \sum_{j=1}^{|\mathcal{S}|} u_i(B_j^i) + (n - |\mathcal{S}|) \cdot x \leq \sum_{j=1}^{|\mathcal{S}|} u_i(B_j^i) + \sum_{j=|\mathcal{S}|+1}^n u_i(B_j^i) = u_i(M \cup D), \end{aligned}$$

where the first inequality holds as each bundle $A_j \notin \mathcal{S}$ that contains divisible goods is valued at most x' to agent i guaranteed by EFM, and the third inequality holds as $u_i(B_j^i) \geq x$ if B_j^i contains no divisible good.

As the permutation is generated uniformly at random, the lemma concludes following the same analysis for ex-ante PROP as in Theorem 4.3:

$$u_i(\mathcal{R}) = \sum_{k=1}^n \sum_{\pi: \pi(k)=i} \frac{u_i(A(\pi, k))}{n!} \geq \sum_{k=1}^n \sum_{\pi: \pi(k)=i} \frac{u_i(B_k^i)}{n!} = \sum_{k=1}^n \frac{u_i(B_k^i)}{n} = \frac{u_i(M \cup D)}{n}. \quad \square$$

Lemma B.9. *The randomized allocation \mathcal{R} is ex-ante PROP for agents in $N \setminus T$.*

Proof. Different from the proof in Lemma B.8 above, a bundle in the actual allocation \mathcal{A} may contain two indivisible goods both valued at b to agent i for some $i \in N \setminus T$. We thus define the baseline allocation for agent i in a different way below.

Definition B.10 (Baseline allocation \mathcal{B}^i for agent $i \in N \setminus T$). The *baseline allocation* \mathcal{B}^i for agent $i \in N \setminus T$ is obtained by

1. letting each bundle contain one distinct good that is matched during the first phases of Steps 2 and 3 (note that there are n goods matched during the first phases of Steps 2 and 3, so each bundle can contain exactly one good);
2. arranging the bundles in descending order according to agent i 's utility;
3. letting agent i allocate remaining indivisible goods by Round-Robin; and
4. executing the water-filling process for the divisible goods according to agent i 's utility.

Assume that agent i is ranked k -th in the permutation π , and denote the actual output allocation by $\mathcal{A} = \{A_1, \dots, A_n\}$, where A_i is also denoted by $A(\pi, k)$ similar to Lemma B.8. The number of the indivisible goods in B_k^i equals to that she receives in the actual allocation before the water-filling process. In the first phase of Step 2, agent i will receive a good with value b , which is no less than the value of the first good added to B_k^i (as the value is at most b). If the second good added to B_k^i is also valued at b to agent i , then the good she receives in the second phase of Step 2 also has value b by Round-Robin. Therefore, if B_k^i contains only indivisible goods, the property $u_i(A_i) \geq u_i(B_k^i)$ holds, as agent i 's value to her bundle only increases during the water-filling process.

We provide a further observation of the indivisible goods in the two allocations before the water-filling process. Denoted by \bar{A}_j and \bar{B}_j^i the j -th bundles that contain only the indivisible goods before the water-filling process for $j \leq n$. The first good added to \bar{A}_j and \bar{B}_j^i is identical. Guaranteed by the second and third steps when constructing the baseline allocation, the possible number of bundles with two goods both with value b to agent i is maximized, i.e., there is no

actual allocation such that the bundles with two goods with value b is strictly larger than that in the baseline allocation. Similarly, subject to the above property, the possible number of bundles with two goods with values a and b respectively is maximized. This allows us to obtain that for any $k' \leq n$, $\sum_{j=1}^{k'} u_i(\bar{B}_j^i) \geq \sum_{\bar{A}_j \in \mathcal{S}} u_i(\bar{A}_j)$ for any set \mathcal{S} of bundles where $|\mathcal{S}| = k'$.

We similarly denote $x = u_i(B_k^i)$ if B_k^i contains a fraction of the divisible good. Denote the set of bundles without any divisible good by \mathcal{S} . From the above analysis, we have $\sum_{j=1}^{k'} u_i(B_j^i) \geq \sum_{\bar{A}_j \in \mathcal{S}} u_i(\bar{A}_j)$. Then, if $u_i(A_i) = x' < x$, we similarly obtain the contradiction that

$$\begin{aligned} u_i(M \cup D) &= \sum_{A_j \in \mathcal{S}} u_i(A_j) + \sum_{A_j \notin \mathcal{S}} u_i(A_j) \leq \sum_{A_j \in \mathcal{S}} u_i(A_j) + (n - |\mathcal{S}|) \cdot x' \\ &< \sum_{j=1}^{|\mathcal{S}|} u_i(B_j^i) + (n - |\mathcal{S}|) \cdot x \leq \sum_{j=1}^{|\mathcal{S}|} u_i(B_j^i) + \sum_{j=|\mathcal{S}|+1}^n u_i(B_j^i) = u_i(M \cup D), \end{aligned}$$

We have shown that $u_i(A(\pi, k)) \geq u_i(B_k^i)$ holds for each π and i such that $\pi(k) = i$. Therefore, for each π , each agent will receive no less value in the actual allocation than in the baseline allocation. Following the same analysis in Lemma B.8, ex-ante PROP is guaranteed. \square

We are now ready to establish Theorem 4.1.

Proof of Theorem 4.1. Lemmas B.8 and B.9 together show the randomized allocation is ex-ante PROP, and it is also ex-post EFM guaranteed by the fact that each partial allocation containing only indivisible goods is EF1 and by Lemma 4.2. Finally, it is straightforward that each part of our algorithm runs in polynomial time. This concludes Theorem 4.1. \square

Remark. The above algorithm can be generalized to the case where agents have personalized bi-valued utilities (the value a and b of each agent are different) to indivisible goods and no constraint to divisible goods.

C Omitted Details in Section 5

C.1 Proof of Lemma 5.3

We can verify the required property after each round and prove this by induction. The initial empty allocation is trivially EFX. (Recall that we use N^r and M^r to denote the corresponding set N and M encountered in Algorithm 2 at the beginning of its round r .) At the next round r , if we cannot find a perfect matching between N^r and goods M^r (lines 13-19), there exists a non-empty minimal unmatchable group Z_t . For each agent $i \in N^r \setminus Z_t$ at this round, since she can still receive a large good and every agent can receive at most one good per round, there is no envy from agent i .

For each agent $i \in Z_t$, if she is frozen in this loop, since she will be only frozen for the next $\lfloor b/a - 1 \rfloor$ rounds, the total utility accumulated by any other agent's bundle during these frozen rounds is upper bounded by $a + \lfloor b/a - 1 \rfloor \cdot a \leq b$, from the second term in Observation 5.2. Thus, each frozen agent will not envy others during the frozen rounds. If an agent $i \in Z_t$ is not frozen (in other words, becomes *quiet*), she will not envy the agents not in Z_t or the agents in Z_t without freezing since these agents can only receive a good in $M^r \setminus \Gamma(Z_t)$, which is valued at a under u_i . For some agent $j \in Z_t$ is frozen in this round, if all goods allocated to j until now are valued at b by agent i , agent i will not envy j after removing the last good from the second term in Observation 5.2. Otherwise, agent i will not envy j since there is exactly one small good in i 's bundle and there is at least one small good in j 's bundle under u_i .

For each agent i not in N^r , i.e., a quiet agent in a previous $Z_{t'}$, it does not envy the agents not in $Z_{t'}$ and the agents in $Z_{t'}$ without freezing before because all agents receive a small good from i 's perspective. Compared to some other quiet agent j in $Z_{t'}$ who has been frozen before, after eliminating one small good in j 's bundle, we have $b \leq a + \lfloor b/a - 1 \rfloor \cdot a + a$, which can eliminate the envy. The term $a + \lfloor b/a - 1 \rfloor \cdot a$ represents the utility during the frozen rounds of j earned by i and the last a represents the utility at this round.

We then come to the second case where we can find a perfect matching between N^r and goods M^r (Lines 10-11). For each agent i in N^r , since it receives large goods at all rounds, there is no envy from her. For each agent i in some previous Z_t , the case is similar to the case above.

From the induction, it suffices to show we can still maintain EFX when allocating the remaining goods at Lines 22-25. Because every agent i does not envy agent j except for one special case as stated in the property we maintain, the envy between these agents can be eliminated by removing the good allocated at this final step. The remaining is to show the case where i and j are in the same Z_t and agent i has never been frozen but agent j has been frozen before. If the value of agent i over the good g allocated to agent j at round r_j is a , the envy can be eliminated after removing the good allocated at this final step since there is no envy from i to j during the loop. Otherwise, j must be before i in the reversed order of π . This is because if i is before j in the reversed order of π , there must exist an augmenting path from i to the good g (from the fact that i can directly take the good g and we keep the remaining allocation to reach a matching with a larger size before the position of j) and then i would be frozen. Thus, i will pick a good before j at this final step from π , which means the envy from i to j can be eliminated after removing the good received by j at this final step (if it exists).

C.2 Proof of Lemma 5.4

For each pair of two agents i and j except for the case that agent i and agent j are in the same Z_t and agent i has never been frozen while agent j has been frozen, from the maintained property as stated before Lemma 5.3, it suffices to show the allocation at the final step (Lines 22-25) can ensure ex-ante EF. If agent i envies agent j in some allocation \mathcal{A}^\approx , either (1) j receives a good g but i does not at the final step or (2) j receives a good g valued b by i but i only receives a small good at the final step. If we compare it to the allocation under π' which only exchanges i and j in π , the envy in the first case can be eliminated because at the final step, j will get nothing and i can receive a good with a weakly larger utility than $u_i(g)$ through an augmenting path, where the worst case is that agent i directly receives g and the remaining allocation is kept.

For the second case (this implies that j is before i in π and j also values g at b), the only thing we need to show is that agent i and j cannot both get a large good from i 's perspective at the final step under π' . Here, we assume i is at position $p^{(i)}$ and j is at position $p^{(j)}$ in π . We have $p^{(i)} > p^{(j)}$. If the above situation happens and assume i and j receive $g^{(i)}$ and $g^{(j)}$ under π' , we can find an augmenting path to let agent i receive a large good under π , which leads to a contradiction. The way to find such a path is as follows.

Since there is a matching for π' by replacing the matching edge (j, g) by (i, g) in π , the size of the maximum matching before position $p^{(i)}$ in π is upper bounded by the size of that before $p^{(i)}$ in π' . If j values $g^{(j)}$ at b , this violates the maximality of the matching up to position $p^{(i)}$ in π since the size should be equal in both maximum matchings up to position $p^{(i)}$ in π and π' . If j values $g^{(j)}$ at a , we then can get the size of the maximum matching before position $p^{(i)}$ in π is exactly equal to the size of that before position $p^{(i)}$ in π' and j values $g^{(i)}$ at b . We then can replace the allocation before position $p^{(i)}$ in π by that before position $p^{(i)}$ in π' and then agent i can receive g'' in π , which violates the maximality of the matching up to position $p^{(i)}$ in π . Both cases lead to a

contradiction, which finishes the proof for this case.

We then come to the exceptional case: agent i and j are in the same Z_t and agent i has never been frozen but agent j has been frozen. If the value of agent i over the good g allocated to j at round r_j is a , there is no envy before the final step and this case can be solved similarly as in the above analysis.

Otherwise, we assume the generated permutation is π . We consider another permutation π' which only exchanges the positions of i and j in π . Since the ex-ante guarantee at the final step has been shown in the above situation, it suffices to show the ex-ante EF is also satisfied before the final step. Because agent j is frozen under π and agent i values the good g at b , agent i will be frozen under π' by directly replacing the agent j in π . If agent i is frozen and agent j is not frozen under π' , then from i 's perspective, the differences between the bundles of agent i and j under π and π' are the same, which ensures the ex-ante EF. It suffices to show that agents i and j cannot be frozen simultaneously under π' .

If both agents i and j are frozen under π' , we can utilize the similar argument over the maximality of the matching before (or up to) the position of i in π to show that i can be also frozen under π , which violates the assumption. This completes the proof.

C.3 Descriptions of Fisher Market

In a Fisher market, we are given a set of n agents denoted by N and a set of m *divisible* goods denoted by M . For each agent $i \in N$, an additive utility function u_i and a budget $e_i \geq 0$ are given. Given a fractional allocation \mathbf{X} with a price vector \mathbf{p} over M , the spending of an agent i under (\mathbf{X}, \mathbf{p}) is $\sum_{g \in M} p_g X_{ig}$. We then define the *maximum bang-per-buck (MBB) ratio* $\alpha_i := \max_{g \in M} u_i(g) / p_g$ and the *MBB-set* $\text{MBB}_i := \{g \in M: u_i(g) / p_g = \alpha_i\}$ for each agent $i \in N$.

A *market equilibrium* (\mathbf{X}, \mathbf{p}) is a fractional allocation \mathbf{X} with a price vector \mathbf{p} over M which satisfies:

- the market clears, i.e. $\sum_{i \in N} X_{ig} = 1$ for each good $g \in M$;
- the budget is fully spent, i.e., $\sum_{g \in M} p_g X_{ig} = e_i$ for each agent $i \in N$;
- each agent $i \in N$ only receives goods in MBB_i .

Theorem C.1 (First Welfare Theorem [Mas-Colell et al., 1995]). *If (\mathbf{X}, \mathbf{p}) is a market equilibrium of a Fisher market, \mathbf{X} is fPO.*

C.4 Proof of Lemma 5.5 when $m < n$

We now come to the corner case when $m < n$. The main difference in this case is the price setting for low-price agents in the above analysis, where it is possible that there exists a low-price agent receiving only a small good. Thus, we need to specify the price separately for this case.

We define the price vector \mathbf{p} in the following way: for each agent i receiving a small good g , we set $p_g = a$. We then set the prices of the remaining goods iteratively: if there exists one agent i without pricing her good g such that some priced agent values g at b , we set $p_g = b$. If there is no further pricing can be made, we set the price as a for each remaining good and call each agent who owns these goods low-price agent, correspondingly. We can then follow a similar proof as the case above to prove this.