A Complete Landscape of EFX Allocations on Graphs: Goods, Chores and Mixed Manna

Yu Zhou[1] , **Tianze Wei**[1] , **Minming Li**[1] and **Bo Li**[2]
[1]Department of Computer Science, City University of Hong Kong, Hong Kong, China
[2]Department of Computing, The Hong Kong Polytechnic University, Hong Kong, China
yzhou567@cityu.edu.hk, t.z.wei-8@my.cityu.edu.hk, minming.li@cityu.edu.hk
comp-bo.li@polyu.edu.hk

Abstract

We study envy-free up to any item (EFX) allocations on graphs where vertices and edges represent agents and items respectively. An agent is only interested in items that are incident to her and all other items have zero marginal values to her. Christodoulou et al. first proposed this setting and studied the case of goods. We extend this setting to the case of mixed manna where an item may be liked or disliked by its endpoint agents. In our problem, an agent has an arbitrary valuation over her incident items such that the items she likes have non-negative marginal values to her and those she dislikes have non-positive marginal values. We provide a complete study of the four notions of EFX for mixed manna in the literature, which differ by whether the removed item can have zero marginal value. We prove that an allocation that satisfies the notion of EFX where the virtually-removed item could always have zero marginal value may not exist and determining its existence is NP-complete, while one that satisfies any of the other three notions always exists and can be computed in polynomial time. We also prove that an orientation (i.e., a special allocation where each edge must be allocated to one of its endpoint agents) that satisfies any of the four notions may not exist, and determining its existence is NP-complete.

1 Introduction

Fair allocation of indivisible items has been broadly studied in the research fields of computer science, economics, and mathematics in the past few decades. One of the most compelling and natural fairness notions is *envy-freeness* (EF), which requires that every agent prefers her own bundle to any other agent's bundle. Though envy-freeness can always be satisfied for divisible items [Aziz and Mackenzie, 2016; Dehghani *et al.*, 2018], it is too demanding for indivisible items. An EF allocation does not exist even for the simple instance where there are two agents and one indivisible item with non-zero marginal values to both agents.

The fact that envy-freeness is hard to satisfy necessitates the study of its relaxations, the most popular one among which is *envy-freeness up to any item* (EFX). EFX requires that any envy could be eliminated by virtually removing any item that the envious agent likes from the envied agent's bundle or any item that the envious agent dislikes from her own bundle. As remarked by [Caragiannis *et al.*, 2019]: *"Arguably, EFX is the best fairness analog of envy-freeness for indivisible items."* Despite significant effort in the literature, the existence of EFX allocations still remains an open problem for indivisible items. Only a few special cases are known to admit EFX allocations [Plaut and Roughgarden, 2020; Chaudhury *et al.*, 2020; Amanatidis *et al.*, 2021; Hosseini *et al.*, 2021; Li *et al.*, 2022].

Recently, [Christodoulou *et al.*, 2023] studied EFX allocations on graphs where vertices correspond to agents and edges correspond to indivisible goods. An agent (vertex) is only interested in the goods (edges) that are incident to her and all other edges have zero marginal values to her. Thus, each good is liked by exactly two agents in their setting. As motivated in [Christodoulou *et al.*, 2023], a direct application of this setting is the allocation of geographical resources, for instance, natural resources among countries on the boundaries, working offices among research groups, and public areas among communities in a region, etc. [Christodoulou *et al.*, 2023] proved that EFX allocations always exist and can be computed in polynomial time for arbitrary graphs. Remarkably, this is one more rare case with more than three agents for which an EFX allocation is guaranteed to exist. They also considered a more restricted scenario where each edge must be allocated to one of its endpoint agents. In this scenario, an allocation is also called an *orientation*. Unfortunately, [Christodoulou *et al.*, 2023] proved that an EFX orientation may not exist, and determining whether it exists or not is NP-complete.

Besides goods, recent years have seen a rapidly growing interest in the case of mixed manna in the literature of fair division [Aziz *et al.*, 2022; Aleksandrov and Walsh, 2019; Liu *et al.*, 2023]. A mixed manna contains items that are goods for some agents but chores for others. Practically, the setting of mixed manna can model the scenarios where agents have different opinions on items. Many real-world scenarios involve the allocation of mixed manna. For example, when the items are paid jobs, they are goods for some people because completing them can bring extra revenue; however, they can be chores for some people who do not care much

	Orientation	Allocation
EFX_0^0	may not exist, NP-c (Corollary 1)	may not exist, NP-c (Theorem 2)
EFX_-^0	may not exist, NP-c (Corollary 1)	always exist, P (Theorem 3)
EFX_0^+	may not exist, NP-c (Corollary 2)	always exist, P (Theorem 4)
EFX_-^+	may not exist, NP-c (Theorem 1)	always exist, P (Corollary 3)

Table 1: Main results. "NP-c" means determining the existence of the corresponding orientations/allocations is NP-complete. "P" means the allocations can be found in polynomial time.

about this amount of money and would like to save time for other matters. The case of mixed manna is also a typical setting where the valuations are not monotone. [Christodoulou *et al.*, 2023]'s graphic nature also appears in the setting of mixed manna. For example, in sports games, each match (that can be viewed as an item) involves two teams (that can be viewed as the agents) and has to be hosted by one of them (i.e., home or away). Hosting a match might be a good for some teams as they can make profit and might be a chore as they cannot cover the expenses. Furthermore, the graph orientation setting can use the topology to indicate who are capable of completing what jobs (edges), so that the jobs can only be allocated to people (incident vertices) who are able to do them. The allocation setting can model the case when people really do not have any cost or benefit on the items they are not incident to.

1.1 Our Problem and Results

In this work, we extend the model of [Christodoulou *et al.*, 2023] to the case of mixed manna, where an edge may be liked or disliked by its endpoint agents. We consider the four variants of EFX for mixed manna in the literature, i.e., EFX_0^0, EFX_-^0, EFX_0^+, and EFX_-^+, where the super and sub scripts indicate the items removed from the envied agent's and the envious agent's bundles respectively, $+/-$ means an item with a strictly positive or negative margin and 0 means an item with a possibly zero margin.

Similar as [Christodoulou *et al.*, 2023], we first study the setting where each edge must be allocated to one of its endpoint agents, i.e., orientations. The main results are summarized in the second column of Table 1. Specifically, we show that an orientation that satisfies any of the four EFX notions may not exist, and determining its existence is NP-complete. Due to the hardness results for orientations, we also study some simple graphs such as trees, stars and paths, for which the existence of orientations that satisfy the four notions can be determined in polynomial time.

We then study the setting where the edges can be allocated to any agent. The main results are summarized in the third column of Table 1. Specifically, we show that an EFX_0^0 allocation may not exist and determining its existence is NP-complete. In contrast, an allocation that satisfies any of the other three notions always exists and can be computed in polynomial time.

1.2 More Related Works

There are many other works that study fair allocation of indivisible items on graphs, whose settings whereas, are quite different from ours and [Christodoulou *et al.*, 2023]'s. [Bouveret *et al.*, 2017] formalized the problem that there is an underlying graph whose vertices are indivisible items and each agent must receive a connected component of the graph. They considered several fairness notions such as proportionality, envy-freeness, maximin share, and gave hardness results for general graphs and polynomial-time algorithms for special graphs. Many following works investigated the same problem with different fairness notions or graph structures [Bilò *et al.*, 2022; Suksompong, 2019; Igarashi and Peters, 2019]. [Bei *et al.*, 2022] considered the same model and quantified the loss of fairness when imposing the connectivity constraint, i.e., *price of connectivity*. [Madathil, 2023] studied a similar model where each agent must receive a compact bundle of items that are "closely related". Different from this line of works, [Hummel and Hetland, 2022] used a graph to reflect conflicts between items. Each vertex on the graph is an item and each edge means that its two endpoint items have a conflict. They require that two items that have a conflict cannot be allocated to the same agent. In other words, the bundle allocated to each agent must be an independent set of the graph. [Payan *et al.*, 2023] studied fair allocation on graph where vertices are agents (as in our setting). The graph was used to relax fairness notions such that fairness only need to be satisfied for the endpoint agents of the edges.

2 Preliminaries

For any positive integer k, let $[k] = \{1, \ldots, k\}$. In an instance of our problem, there is a graph $G = (N, M)$ where $N = \{a_1, \ldots, a_n\}$ is the vertex set and M is the edge set. Each vertex corresponds to an agent and each edge corresponds to an indivisible item. We use vertex and agent, edge and item, interchangeably. We also write both (a_i, a_j) and $e_{i,j}$ to represent the edge between a_i and a_j. Each agent $a_i \in N$ has a valuation $v_i : 2^M \to \mathbb{R}$ over the edges and $v_i(\emptyset) = 0$. We also write $v_i(e)$ to represent $v_i(\{e\})$.

For an agent a_i, each item $e \in M$ is classified as a *good* (if it has strictly positive marginal values to a_i, i.e., $v_i(S \cup \{e\}) > v_i(S)$ for any $S \subseteq M \setminus \{e\}$), a *chore* (if it has strictly negative marginal values to a_i, i.e., $v_i(S \cup \{e\}) < v_i(S)$ for any $S \subseteq M \setminus \{e\}$), or a *dummy* (if it has zero marginal value to a_i, i.e., $v_i(S \cup \{e\}) = v_i(S)$ for any $S \subseteq M \setminus \{e\}$). Accordingly, an instance is called a *goods instance* (if no item is a chore for any agent), a *chores instance* (if no item is a good for any agent), or a *mixed instance* (if an item may be a good, a chore, or a dummy for any agent). Let E_i be the set of all edges that are incident to a_i, $E_i^{\geq 0} \subseteq E_i$ be the subset of non-chores for a_i, and $E_i^{>0} \subseteq E_i$ be the subset of goods for a_i. Note that in our setting, all edges that are not incident to a_i (i.e., $M \setminus E_i$) are dummies for a_i.

An *allocation* $\mathbf{X} = (X_1, \ldots, X_n)$ is an n-partition of M such that X_i contains the edges allocated to agent $a_i \in N$, where $X_i \cap X_j = \emptyset$ for any $a_i, a_j \in N$ and $\bigcup_{a_i \in N} X_i = M$. An *orientation* is a restricted allocation where each edge must be allocated to one of its endpoint agents. An allocation $\mathbf{X}$ is partial if $\bigcup_{a_i \in N} X_i \subsetneq M$.

2.1 Fairness Notions

Given an allocation $\mathbf{X}$, we say agent a_i envies agent a_j if $v_i(X_j) > v_i(X_i)$. The allocation is *envy-free* (EF) if no agent envies the others, i.e., for every two agents $a_i, a_j \in N$, $v_i(X_i) \geq v_i(X_j)$. As we have seen, envy-freeness is too demanding for indivisible items. Thus, in this paper, we focus on its relaxation *envy-free up to any item* (EFX).

For the case of mixed manna, there are four variants of EFX in the literature [Aziz *et al.*, 2022; Aleksandrov and Walsh, 2020; Bérczi *et al.*, 2020], namely, EFX_0^0, EFX_-^0, EFX_0^+, and EFX_-^+. EFX_0^0 requires that any envy could be eliminated by removing any item that is not a chore for the envious agent from the envied agent's bundle or any item that is not a good from the envious agent's own bundle. Formally,

Definition 1 (EFX_0^0). *An allocation $\mathbf{X} = (X_1, \dots, X_n)$ is EFX_0^0 if for every two agents $a_i, a_j \in N$ such that a_i envies a_j, both of the following conditions hold:*

1. *for any $e \in X_j$ such that $v_i(X_j \setminus \{e\}) \leq v_i(X_j)$, $v_i(X_i) \geq v_i(X_j \setminus \{e\})$;*
2. *for any $e \in X_i$ such that $v_i(X_i \setminus \{e\}) \geq v_i(X_i)$, $v_i(X_i \setminus \{e\}) \geq v_i(X_j)$.*

EFX_-^0 differs from EFX_0^0 in that the item removed from the envious agent's bundle cannot be a dummy. More concretely, the item e considered in the second condition is subject to $v_i(X_i \setminus \{e\}) > v_i(X_i)$. EFX_0^+ differs from EFX_0^0 in that the item removed from the envied agent's bundle cannot be a dummy, i.e., the item e considered in the first condition is subject to $v_i(X_j \setminus \{e\}) < v_i(X_j)$. EFX_-^+ differs from EFX_0^0 in that the item removed from the envied agent's and the envious agent's bundles cannot be a dummy. The formal definitions of EFX_-^0, EFX_0^+ and EFX_-^+ can be seen in the full version.

Obviously, any EFX_0^0 allocation is also EFX_-^0 or EFX_0^+, and any EFX_-^0 or EFX_0^+ allocation is also EFX_-^+.

Goods and Chores Instances. Goods instances have been well studied in [Christodoulou *et al.*, 2023], and we will see that our results provide alternative approaches. For chores instances, we provide a discussion in the full version of this paper. In the subsequent sections, we shall focus on the general case of mixed manna.

3 EFX Orientations

In this section, we elaborate on EFX orientations. Firstly, we have the following proposition, whose proof can be seen in the full version of this paper.

Proposition 1. *There exist graphs for which no orientation satisfies any of the four notions of EFX.*

Due to this negative result, we turn to studying the complexity of determining the existence of EFX orientations. The result by [Christodoulou *et al.*, 2023] (see Theorem 2 in their paper) directly implies that determining the existence of EFX_-^0 orientations is NP-complete. In the graphs constructed in their reduction, each edge is a good for both its endpoint agents. For such graphs, any EFX_-^0 orientation is also EFX_0^0. Therefore, we have the following corollary.

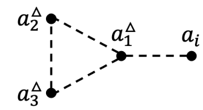

Figure 1: A gadget where agent a_i must receive (a_i, a_1^Δ) if the orientation is EFX_-^+. Each dashed edge is a chore for both its endpoints.

Corollary 1. *Determining whether an EFX_0^0 or EFX_-^0 orientation exists or not is NP-complete.*

In the following, we prove the below theorem for EFX_-^+.

Theorem 1. *Determining whether an EFX_-^+ orientation exists or not is NP-complete, even for additive valuations*[1].

To prove Theorem 1, we reduce from $(3, B2)$-SAT problem to the EFX_-^+ orientation problem. A $(3, B2)$-SAT instance contains a Boolean formula in conjunctive normal form consisting of n variables $\{x_i\}_{i \in [n]}$ and m clauses $\{C_j\}_{j \in [m]}$. Each variable appears exactly twice as a positive literal and exactly twice as a negative literal in the formula, and each clause contains three distinct literals. Determining whether a $(3, B2)$-SAT instance is satisfiable or not is NP-complete [Berman *et al.*, 2007].

Our reduction uses a gadget to ensure that a specific agent must receive a chore if the orientation is EFX_-^+. One such gadget is shown in Figure 1. In this example, agent a_i must receive (a_i, a_1^Δ) if the orientation is EFX_-^+. Otherwise, one of the other three agents must receive at least two chores and envy a_i even after removing one chore.

Given a $(3, B2)$-SAT instance $(\{x_i\}_{i \in [n]}, \{C_j\}_{j \in [m]})$, we construct a graph as follows:

- For each variable x_i, create two vertices a_i^T, a_i^F and one edge (a_i^T, a_i^F) with a value of 2 to both a_i^T and a_i^F.
- For each clause C_j, create one vertex a_j^C. Besides, if C_j contains a positive literal x_i, create one edge (a_j^C, a_i^T) with a value of 1 to both a_j^C and a_i^T. If C_j contains a negative literal $\neg x_i$, create one edge (a_j^C, a_i^F) with a value of 1 to both a_j^C and a_i^F.
- Create three vertices $a_1^\Delta, a_2^\Delta, a_3^\Delta$ and three edges $(a_1^\Delta, a_2^\Delta), (a_2^\Delta, a_3^\Delta), (a_1^\Delta, a_3^\Delta)$. Besides, for each $i \in [n]$, create two edges (a_i^T, a_1^Δ) and (a_i^F, a_1^Δ). For each $j \in [m]$, create one edge (a_j^C, a_1^Δ). Each of these edges has a value of -1 to both its endpoint agents.
- Each vertex has an additive valuation.

To visualize the above reduction, we show the graph constructed from the formula $(x_1 \vee x_2 \vee x_3) \wedge (x_1 \vee x_2 \vee \neg x_3) \wedge (\neg x_1 \vee \neg x_2 \vee \neg x_3) \wedge (\neg x_1 \vee \neg x_2 \vee x_3)$ in Figure 2.

We now prove that a $(3, B2)$-SAT instance is satisfiable if and only if the constructed graph has an EFX_-^+ orientation.

[1] Valuation v_i is additive if $v_i(S) = \sum_{e \in S} v_i(e)$ for any $S \subseteq M$.

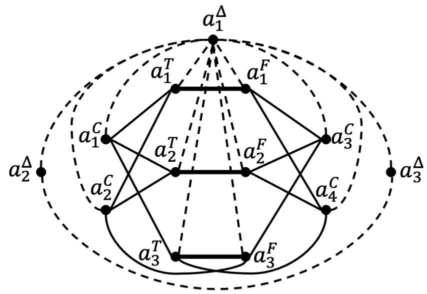

Figure 2: The graph constructed from the formula $(x_1 \vee x_2 \vee x_3) \wedge (x_1 \vee x_2 \vee \neg x_3) \wedge (\neg x_1 \vee \neg x_2 \vee \neg x_3) \wedge (\neg x_1 \vee \neg x_2 \vee x_3)$, where each edge has the same value to both its endpoint agents, each bold solid edge has a value of 2, each non-bold solid edge has a value of 1 and each dashed edge has a value of -1.

Proof of Theorem 1. For ease of presentation, for each variable x_i, we denote by $C_{j^{i,T,1}}, C_{j^{i,T,2}}$ the two clauses that contain the positive literal x_i and by $C_{j^{i,F,1}}, C_{j^{i,F,2}}$ the two clauses that contain the negative literal $\neg x_i$.

For one direction, we assume that the $(3, B2)$-SAT instance has a satisfying assignment and use the assignment to create an EFX$^+_-$ orientation as follows:

- Allocate (a_1^Δ, a_2^Δ) to a_1^Δ, (a_2^Δ, a_3^Δ) to a_2^Δ, and (a_3^Δ, a_1^Δ) to a_3^Δ. Allocate each other edge that is incident to a_1^Δ to the endpoint that is not a_1^Δ.
- For each variable x_i that is set to True, allocate (a_i^T, a_i^F) to a_i^T, $(a_i^T, a^C_{j^{i,T,1}})$ to $a^C_{j^{i,T,1}}$, $(a_i^T, a^C_{j^{i,T,2}})$ to $a^C_{j^{i,T,2}}$, and $(a_i^F, a^C_{j^{i,F,1}})$, $(a_i^F, a^C_{j^{i,F,2}})$ to a_i^F.
- For each variable x_i that is set to False, allocate (a_i^T, a_i^F) to a_i^F, $(a_i^F, a^C_{j^{i,F,1}})$ to $a^C_{j^{i,F,1}}$, $(a_i^F, a^C_{j^{i,F,2}})$ to $a^C_{j^{i,F,2}}$, and $(a_i^T, a^C_{j^{i,T,1}})$, $(a_i^T, a^C_{j^{i,T,2}})$ to a_i^T.

Next, we show that the above orientation is EFX$^+_-$. For agents $a_1^\Delta, a_2^\Delta, a_3^\Delta$, each of them receives one edge with a value of -1 and all edges received by other agents have non-positive values to them. After removing the edge from their bundles, they do not envy others. For each variable x_i that is set to True, agent a_i^T does not envy others since she receives a total value of 1 and each of her incident edges that she does not receive has a value of 1. Agent a_i^F receives three edges with values of $1, 1, -1$, respectively. The only incident edge that she does not receive is (a_i^T, a_i^F), which is allocated to a_i^T and has a value of 2. After removing the edge with a value of -1 from her own bundle or (a_i^T, a_i^F) from a_i^T's bundle, a_i^F does not envy a_i^T. We have an analogous argument for each variable that is set to False. It remains to consider the agents that correspond to clauses. Since the assignment is satisfying, each clause contains at least one literal that is evaluated to True. This implies that each agent a_j^C receives at least one edge with a value of 1. For example, if the clause C_j contains a positive literal x_i that is evaluated to True, a_j^C receives the edge (a_i^T, a_j^C). Since each of a_j^C's incident edges that she does not receive has a value of 1, a_j^C does not envy other agents after removing the edge with a value of -1 from her own bundle or the edge with a value of 1 from other agents' bundles.

For the other direction, we assume that the constructed graph has an EFX$^+_-$ orientation and use the orientation to create a satisfying assignment as follows: for each variable x_i, if the edge (a_i^T, a_i^F) is allocated to agent a_i^T, then set x_i to True; otherwise, set x_i to False. Next, we show that the assignment is satisfying. First, since the orientation is EFX$^+_-$, each agent that corresponds to a variable or a clause must receive the edge between herself and a_1^Δ that has a value of -1. For each variable x_i, if the edge (a_i^T, a_i^F) is allocated to agent a_i^T, both $(a_i^F, a^C_{j^{i,F,1}})$ and $(a_i^F, a^C_{j^{i,F,2}})$ must be allocated to agent a_i^F. Otherwise, a_i^F will envy a_i^T even after removing (a_i^F, a_1^Δ) from her own bundle. For a similar reason, if the edge (a_i^T, a_i^F) is allocated to a_i^F, both $(a_i^T, a^C_{j^{i,T,1}})$ and $(a_i^T, a^C_{j^{i,T,2}})$ must be allocated to a_i^T. For each clause C_j, agent a_j^C must receive at least one edge with a value of 1. Otherwise, a_j^C will envy the agents who receive her incident edges that have a value of 1 even after removing the edge with a value of -1 from her own bundle. This implies that each clause has a literal that is evaluated to True. $\square$

Notice that in the graphs constructed in the above reduction, each edge has non-zero values to both its endpoint agents. For such graphs, an orientation is EFX$^+_0$ if and only if it is EFX$^+_-$, since no agent receives an edge with a value of zero. Therefore, the hardness of determining the existence of EFX$^+_-$ orientations also applies to EFX$^+_0$ orientations.

Corollary 2. *Determining whether an EFX$^+_0$ orientation exists or not is NP-complete, even for additive valuations.*

Simple Graphs. To bypass the hardness results in Corollaries 1, 2 and Theorem 1 for general graphs, in the full version of this paper, we also study EFX orientations on some simple graphs such as trees, stars and paths. For these simple graphs, though orientations that satisfy the four notions may not exist, their existence can be determined in polynomial time.

4 EFX Allocations

In this section, we elaborate on EFX allocations.

4.1 EFX$^0_0$ Allocations

We start with the strongest one among those four notions, i.e., EFX$^0_0$. We say an edge e is *priceless* to an agent a_i if for any $S_1, S_2 \subseteq M$ such that $e \notin S_1$ and $e \in S_2$, we have $v_i(S_1) < v_i(S_2)$. We first have the following proposition, which provides some characterization of EFX$^0_0$ allocations on some graphs with priceless edges. The proof can be seen in the full version of this paper.

Proposition 2. *For graphs that satisfy (1) each edge is a good for both its endpoint agents, (2) each agent has one priceless incident edge and (3) each priceless edge is priceless to both its endpoint agents, we have that each edge must be allocated to one of its endpoint agents in any EFX$^0_0$ allocation.*

In the full version, we provide a graph with priceless edges, which proves the following proposition.

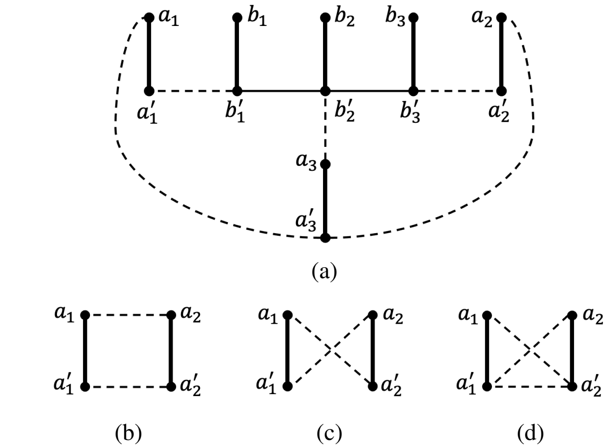

Figure 3: (a) OR gadget, (b) NOT gadget, (c) WIRE gadget, (d) TRUE terminator gadget. In these graphs, each agent has an additive valuation. Each bold solid edge is priceless to both its endpoint agents (e.g., it has an infinitely large value of $+\infty$), each non-bold solid edge has an infinitely small value of $\epsilon_1 > 0$ to both its endpoint agents, each dashed line also has an infinitely small value of ϵ_2 to both its endpoint agents with $\epsilon_1 > \epsilon_2 > 0$.

Proposition 3. *There exist graphs for which no allocation is EFX_0^0.*

We next study the complexity of determining the existence of EFX_0^0 allocations and have the following result.

Theorem 2. *Determining whether an EFX_0^0 allocation exists or not is NP-complete, even for additive valuations.*

To prove Theorem 2, we reduce from Circuit-SAT problem to the EFX_0^0 allocation problem. Circuit-SAT problem determines whether a given Boolean circuit has an assignment of the inputs that makes the output True, which is well-known to be NP-complete [Karp *et al.*, 1975].

We first show how to simulate the OR gate, the NOT gate, the wire in the circuit and how to force the final output to be True. To achieve this, we construct four graphs, named OR gadget, NOT gadget, WIRE gadget, TRUE terminator gadget, respectively (see Figure 3). It is easy to see that Proposition 2 applies to all these four gadgets. That is, in any EFX_0^0 allocation on each of these gadgets, each edge must be allocated to one of its endpoint agents. This enables us to represent each input (or output) in the circuit as an edge in the gadgets and its value (True or False) as the orientation of the edge.

In the OR gadget (see Figure 3a), edges (a_1, a'_1) and (a_2, a'_2) represent the two inputs of the OR gate, edge (a_3, a'_3) represents the output. The following claim shows that the OR gadget correctly simulates the OR gate.

Claim 1. *In every EFX_0^0 allocation on the OR gadget, edge (a_3, a'_3) is allocated to a_3 if and only if edge (a_1, a'_1) is allocated to a_1 or edge (a_2, a'_2) is allocated to a_2.*

Proof. We first show that if (a_1, a'_1) is allocated to a_1, (a_3, a'_3) must be allocated to a_3. Since (a_1, a'_1) is priceless to a'_1 but is allocated to a_1, a'_1 envies a_1. Hence, (a_1, a'_3) must be allocated to a'_3. Otherwise, a'_1 still envies a_1 after removing (a_1, a'_3) from a_1's bundle. Moreover, (a_3, a'_3) must be allocated to a_3. Otherwise, a_3 still envies a'_3 after removing (a_1, a'_3) from a_3's bundle. By symmetry, it holds that if (a_2, a'_2) is allocated to a_2, (a_3, a'_3) must be allocated to a_3.

We then show that when (a_1, a'_1) is allocated to a_1 and (a_3, a'_3) is allocated to a_3, no matter which endpoint agent (a_2, a'_2) is allocated to, there exists an EFX_0^0 allocation. When (a_2, a'_2) is allocated to a_2, we construct an EFX_0^0 allocation as follows: allocate each priceless edge to the upper endpoint agent, i.e., (a_i, a'_i) to a_i for every $i \in \{1, 2, 3\}$ and (b_i, b'_i) to b_i for every $i \in \{1, 2, 3\}$; allocate the middle four edges to the endpoint agents who are further away from b'_2, i.e., (a'_1, b'_1) to a'_1, (b'_1, b'_2) to b'_1, (b'_2, b'_3) to b'_3, (b'_3, a'_2) to a'_2; allocate (b'_2, a_3) to b'_2, (a_1, a'_3) to a'_3, (a_2, a'_3) to a'_3. Since each agent has a positive value for each edge she receives, to verify that the allocation is EFX_0^0, it suffices to consider the agents who receive more than one edge (only a'_3 in the above allocation). Since both a_1 and a_2 receive their priceless edges, neither of them envies a'_3 and thus the allocation is EFX_0^0. When (a_2, a'_2) is allocated to a'_2, we construct an EFX_0^0 allocation as follows: allocate each priceless edge except (a_2, a'_2) and (b_1, b'_1) to the upper endpoint, i.e., (a_i, a'_i) to a_i for every $i \in \{1, 3\}$, (b_i, b'_i) to b_i for every $i \in \{2, 3\}$, (a_2, a'_2) to a'_2, (b_1, b'_1) to b'_1; for the middle four edges, allocate (a'_1, b'_1) to a'_1, (b'_1, b'_2) to b'_2, (b'_2, b'_3) to b'_3, (b'_3, a'_2) to b'_3; allocate (b'_2, a_3) to b'_2, (a_1, a'_3) to a'_3, (a_2, a'_3) to a_2. In the above allocation, only b'_2 and b'_3 receive more than one edge. For b'_2, neither b'_1 nor a_3 envies her since both of them receive their priceless edges. For b'_3, a'_2 does not envy her since she receives her priceless edge, and b'_2 does not envy her since she receives a value of $\epsilon_1 + \epsilon_2$ and thinks that b'_3 receives a value of ϵ_1. Therefore, the allocation is also EFX_0^0. By symmetry, when (a_2, a'_2) is allocated to a_2 and (a_3, a'_3) is allocated to a_3, no matter which endpoint agent (a_1, a'_1) is allocated to, there exists an EFX_0^0 allocation.

We next show that if both (a_1, a'_1) and (a_2, a'_2) are allocated to their lower endpoint agents, (a_3, a'_3) must be allocated to a'_3. It suffices to show that (b'_2, a_3) must be allocated to a_3. This is because if both (a_3, a'_3) and (b'_2, a_3) are allocated to a_3, a'_3 will envy a_3 even after removing (b'_2, a_3) from a_3's bundle. If (b_2, b'_2) is allocated to b'_2, (b'_2, a_3) must be allocated to a_3 and we have done, since otherwise b_2 will envy b'_2 even after removing (b'_2, a_3) from b'_2's bundle. Therefore, it remains to consider the case when (b_2, b'_2) is allocated to b_2. Since (a_1, a'_1) is allocated to a'_1, (a'_1, b'_1) must be allocated to b'_1 since otherwise a_1 will envy a'_1 even after removing (a'_1, b'_1) from a'_1's bundle. Furthermore, (b_1, b'_1) must be allocated to b_1. By the same reasoning, (a'_2, b'_3) must be allocated to b'_3 and (b_3, b'_3) must be allocated to b_3. Then consider the incident edges of b'_2 that have not been allocated so far, i.e., (b'_1, b'_2) and (b'_2, b'_3). b'_2 must receive one of these two edges, since otherwise she will envy b'_1 even after removing (a'_1, b'_1) from b'_1's bundle, and b'_3 even after removing (a'_2, b'_3) from b'_3's bundle. No matter which edge b'_2 receives, (b'_2, a_3) must be allocated to a_3. To see this, let the edge that b'_2 receives be (b'_1, b'_2). Since b'_1 receives a value of ϵ_2 and thinks that b'_2 receives a value of $\epsilon_1 > \epsilon_2$, she envies b'_2 and thus b'_2 cannot receive (b'_2, a_3) any more.

Lastly, we show that when all of (a_1, a'_1), (a_2, a'_2) and

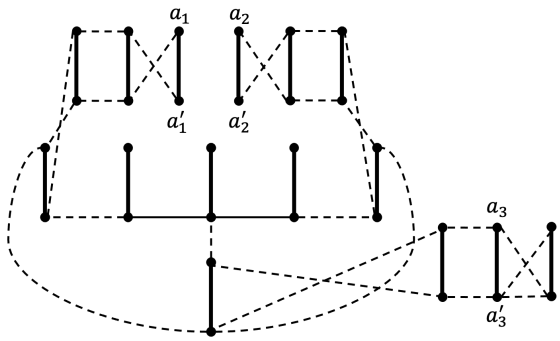

Figure 4: The graph is constructed from the circuit that consists of only one AND gate, two inputs, and one final output. (a_1, a'_1) and (a_2, a'_2) simulate the inputs, (a_3, a'_3) simulates the final output.

(a_3, a'_3) are allocated to their lower endpoint agents, there exists an EFX^0_0 allocation. We allocate each priceless edge except (b_1, b'_1) and (b_3, b'_3) to the lower endpoint agent, i.e., (a_i, a'_i) to a'_i for every $i \in \{1, 2, 3\}$, (b_2, b'_2) to b'_2, and (b_i, b'_i) to b_i for every $i \in \{1, 3\}$; for the middle four edges, allocate (a'_1, b'_1) and (b'_1, b'_2) to b'_1, (b'_2, b'_3) and (b'_3, a'_2) to b'_3; allocate (b'_2, a_3) to a_3, (a_1, a'_3) to a_1, (a_2, a'_3) to a_2. In the above allocation, only b'_1 and b'_3 receive more than one edge. For b'_1, neither a'_1 nor b'_2 envies her since both of them receive their priceless edges. By the same reasoning, neither b'_2 nor a'_2 envies b'_3. Therefore, the allocation is EFX^0_0. □

We can also prove that the other three gadgets correctly simulate the corresponding elements of a circuit. The formal claims and proofs can be seen in the full version of this paper.

Given a circuit, we first substitute each AND gate with three NOT gates and one OR gate, and get an equivalent circuit without AND gates. For the new circuit, we construct a priceless edge with a value of $+\infty$ for each input, and the corresponding gadget for each gate and wire. We then construct a True terminator gadget to force the final output to be True. Figure 4 shows the graph constructed from a simple circuit with one AND gate, two inputs and one final output. Note that Proposition 2 still applies to the graph we construct.

Up to now, it is not hard to see the correctness of Theorem 2, whose formal proof can be seen in the full version.

Remark 1. *Our reduction borrows an idea from the reduction by [Christodoulou* et al.*, 2023] (see Theorem 2 in their paper) and generalizes their reduction. Our reduction can imply their result, while theirs cannot carry over to our problem since it relies on the orientation model.*

4.2 EFX^0_- Allocations

We next study EFX^0_- and have the following theorem.

Theorem 3. *For any graph, an EFX^0_- allocation always exists and can be computed in polynomial time.*

We first introduce some notations. Given a (partial) allocation $\mathbf{X} = (X_1, \ldots, X_n)$, let $R(\mathbf{X})$ denote the set of unallocated edges, i.e., $R(\mathbf{X}) = M \setminus \bigcup_{a_i \in N} X_i$. We say an agent a_j is *safe* for another agent a_i if a_i does not envy a_j even if a_j receives all her unallocated incident edges that are not chores for her, i.e., $v_i(X_i) \geq v_i(X_j \cup (E_i^{\geq 0} \cap R(\mathbf{X})))$. We next introduce some properties of allocations.

Definition 2 (Properties of a (Partial) Allocation)**.** *We say that a (partial) allocation $\mathbf{X}$ satisfies*

- *property (1) if for every agent a_i, the value of her bundle is at least the largest value among her unallocated incident edges that are not chores for her. That is, $v_i(X_i) \geq v_i(e)$ for every edge $e \in E_i^{\geq 0} \cap R(\mathbf{X})$;*
- *property (2) if for every envied agent a_i, the value of her bundle is at least the value of all her unallocated incident edges that are not chores for her. That is, $v_i(X_i) \geq v_i(E_i^{\geq 0} \cap R(\mathbf{X}))$;*
- *property (3) if for every two envied agents, there exists a non-envied agent who is safe for both of them;*
- *property (4) if no agent receives an edge that is a chore for her. That is, $e \in E_i^{\geq 0}$ for any $a_i \in N$ and $e \in X_i$;*
- *property (5) if every envied agent a_i receives exactly one edge, i.e., $|X_i| = 1$.*
- *property (6) if every envied agent is envied by exactly one agent;*
- *property (7) if there is no envy cycle among the agents. That is, there does not exist a sequence of the agents $a_{i_0} \leftarrow a_{i_1} \leftarrow \cdots \leftarrow a_{i_s}$ such that a_{i_l} envies $a_{i_{l-1}}$ for every $l \in [s]$ and $i_0 = i_s$;*
- *property (8) if for any sequence of agents $a_{i_0} \leftarrow a_{i_1} \leftarrow \cdots \leftarrow a_{i_s}$ such that a_{i_l} envies $a_{i_{l-1}}$ for every $l \in [s]$ and a_{i_s} is non-envied, we have that a_{i_l} is safe for a_{i_0} for every $l \in [s]$.*

We obtain an EFX^0_- allocation in two parts.

Part 1. In the first part, we compute a (partial) EFX^0_- orientation that satisfies properties (1)-(8) in Definition 2. Our algorithms in this part are adapted from those by [Christodoulou *et al.*, 2023]. There are two differences between our algorithms and [Christodoulou *et al.*, 2023]'s. First, since there is one more requirement in our problem that agents cannot envy others after removing a chore from their own bundles, we need to carefully allocate the edges that are chores for their endpoint agents. Second, the algorithms by [Christodoulou *et al.*, 2023] cannot guarantee property (8) and our algorithms need to deal with the case where property (8) is not satisfied.

We have the following lemma. The detailed algorithms and proofs can be seen in the full version.

Lemma 1. *For any graph, a (partial) EFX^0_- orientation that satisfies properties (1)-(8) in Definition 2 can be computed in polynomial time.*

Part 2. In the second part, we allocate the edges that are not allocated in Part 1. We first categorize the unallocated edges into four disjoint groups:

- G_1 contains each edge that has at least one non-envied endpoint agent for whom the edge is not a chore;
- G_2 contains each edge that has two envied endpoints;
- G_3 contains each edge that has one non-envied endpoint agent for whom the edge is a chore and one envied endpoint agent for whom it is not a chore;

- G_4 contains the edges that have not been included in G_1, G_2, G_3. Notice that each edge in G_4 is a chore for both its endpoint agents.

We will allocate the unallocated edges from G_1 to G_4 such that no agent will receive an edge that is a chore for her and thus property (4) will be retained. Besides, no agent will get worse off and no allocated edge will become unallocated, which will ensure that properties (1) and (2) are retained. Moreover, no new envy will occur, which will ensure that properties (6) and (7) are retained. Furthermore, no allocated edge will be reallocated to another agent, which will ensure that an agent who is safe for some agent is always safe for that agent and thus properties (3) and (8) are retained. We will also see that the (partial) allocation is always EFX^0_- during the allocation process. Specifically,

- For each edge in G_1, we allocate it to the non-envied endpoint agent for whom it is not a chore.
- For each edge in G_2, we allocate it to the non-envied agent who is safe for both its endpoint agents.
- For each edge $e_{i,j}$ in G_3, we consider three cases. Without loss of generality, let a_i be the endpoint agent for whom $e_{i,j}$ is not a chore and a_j be the other one for whom it is a chore. First, a_i becomes non-envied. Similar to the allocation of G_1, we allocate the edge to a_i. Second, there exists a non-envied agent $a_k \neq a_j$ who is safe for a_i. Similar to the allocation of G_2, we allocate $e_{i,j}$ to a_k. Third, a_j is the only non-envied agent who is safe for a_i. By property (8), it must be the case that there exists a sequence of agents $a_{i_0} \leftarrow a_{i_1} \leftarrow \cdots \leftarrow a_{i_s}$ such that a_{i_l} envies $a_{i_{l-1}}$ for every $l \in [s]$, a_{i_0} is a_i and a_{i_s} is a_j. For this case, we allocate $e_{i,j}$ to $a_{i_{s-1}}$.
- For G_4, we consider two cases. First, if no agent is envied, we allocate each edge to an agent who is not its endpoint. Second, if some agent is envied, we find two agents a_i and a_j such that a_i is envied by a_j and a_j is non-envied. We allocate the edges in G_4 that are incident to a_j (i.e., $E_j \cap G_4$) to a_i, and the other edges in G_4 (i.e., $G_4 \setminus E_j$) to a_j.

We have the following lemma in Part 2, whose proof can be seen in the full version of this paper.

Lemma 2. *For any graph, given a (partial) EFX^0_- orientation that satisfies properties (1)-(8) in Definition 2, we can compute an EFX^0_- allocation in polynomial time.*

By Lemmas 1 and 2, it is clear that Theorem 3 holds.

4.3 EFX^+_0 Allocations

Finally, we study EFX^+_0 and have the following theorem.

Theorem 4. *For any graph, an EFX^+_0 allocation always exists and can be computed in polynomial time.*

First recall that for chores instances where each edge is a chore for both its endpoint agents, we can compute an envy-free allocation by allocating each edge to an agent who is not its endpoint. Thus in the following, we only consider graphs where there exists an edge that is not a chore for at least one of its endpoint agents.

To get some intuitions about how to compute an EFX^+_0 allocation, consider the graphs where each edge is a good for at least one of its endpoint agents. For these graphs, we can simply allocate each edge to the endpoint agent for whom it is a good. For any agent a_i, each edge she receives is a good for her, and at most one edge that each other agent a_j receives is a good for her. After removing the good from a_j's bundle, a_i does not envy a_j. Thus, the allocation is EFX^+_0.

The trickier graphs to deal with are those with edges that are not goods for any of their endpoint agents. For these graphs, we want to find an agent who can receive all such edges, so that we can simply allocate each remaining edge to one of its endpoint agents as above. At the same time, the allocation should be EFX^+_0 for the agent we find. When there exists an agent a_i to whom the total value of her incident edges is non-negative (i.e., $v_i(E_i) \geq 0$), we let a_i receive all her incident edges as well as all edges that are not goods for any of their endpoint agents. We then allocate each remaining edge to one of its endpoint agents for whom it is a good. Since a_i receives all her incident edges whose total value is non-negative, the allocation is EFX^+_0 for her.

However, when the total value of the incident edges is negative to every agent (i.e., $v_i(E_i) < 0$ for every a_i), we cannot simply allocate all incident edges to an agent as above, since the allocation may not be EFX^+_0 for her. For this case, we let an agent receive all her incident edges that are not chores for her and allocate her other incident edges to another agent. More concretely, we first choose an edge $e_{i,j}$ that is not a chore for a_i, breaking the tie by giving priority to the edges that are not chores for one endpoint agent and are chores for the other. We then let a_i receive all her incident edges that are not chores for her, as well as all edges that are not incident to her but are not goods for any of their endpoint agents. Next, we let a_j receive all her unallocated incident edges that are goods for her, as well as a_i's unallocated incident edges that are not goods for any of their endpoint agents. At last, we allocate each remaining unallocated edge to one of its endpoint agents for whom it is a good. The formal description of the allocation process and the formal proof of Theorem 4 can be seen in the full version.

Since any EFX^0_- or EFX^+_0 allocation is also EFX^+_-, we have the following corollary.

Corollary 3. *For any graph, an EFX^+_- allocation always exists and can be computed in polynomial time.*

5 Conclusion

In this paper, we give a complete computational study of EFX allocations on graphs when the items are a mixture of goods and chores. There are some future directions. In our setting, exactly two agents are interested in one common item that is incident to both of them. One immediate direction is to study the generalized setting with multi-edges where multiple edges exist between two agents or hypergraphs where more than two agents are interested in one common item. Another direction is to study the setting where agents are also interested in the edges that are not very far away from them. To bypass the hardness results of EFX orientations, we have studied some simple graphs including trees, stars and paths. One can also study complex graphs for which the existence of EFX orientations can be determined in polynomial time.

Acknowledgements

This work is partly funded by the Hong Kong SAR Research Grants Council (No. PolyU 15224823) and Department of Science and Technology of Guangdong Province (No. 2023A1515010592).

References

[Aleksandrov and Walsh, 2019] Martin Aleksandrov and Toby Walsh. Greedy algorithms for fair division of mixed manna. *CoRR*, abs/1911.11005, 2019.

[Aleksandrov and Walsh, 2020] Martin Aleksandrov and Toby Walsh. Two algorithms for additive and fair division of mixed manna. In *Proceedings of the 43rd German Conference on Artificial Intelligence*, pages 3–17, 2020.

[Amanatidis *et al.*, 2021] 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.

[Aziz and Mackenzie, 2016] Haris Aziz and Simon Mackenzie. A discrete and bounded envy-free cake cutting protocol for any number of agents. In *Proceedings of the 57th IEEE Symposium on Foundations of Computer Science*, pages 416–427, 2016.

[Aziz *et al.*, 2022] Haris Aziz, Ioannis Caragiannis, Ayumi Igarashi, and Toby Walsh. Fair allocation of indivisible goods and chores. *Autonomous Agents and Multi-Agent Systems*, 36(1):3, 2022.

[Bei *et al.*, 2022] Xiaohui Bei, Ayumi Igarashi, Xinhang Lu, and Warut Suksompong. The price of connectivity in fair division. *SIAM J. Discret. Math.*, 36(2):1156–1186, 2022.

[Bérczi *et al.*, 2020] Kristóf Bérczi, Erika R. Bérczi-Kovács, Endre Boros, Fekadu Tolessa Gedefa, Naoyuki Kamiyama, Telikepalli Kavitha, Yusuke Kobayashi, and Kazuhisa Makino. Envy-free relaxations for goods, chores, and mixed items. *CoRR*, abs/2006.04428, 2020.

[Berman *et al.*, 2007] Piotr Berman, Marek Karpinski, and Alexander D. Scott. Computational complexity of some restricted instances of 3-sat. *Discrete Applied Mathematics*, 155(5):649–653, 2007.

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

[Bouveret *et al.*, 2017] Sylvain Bouveret, Katarína Cechlárová, Edith Elkind, Ayumi Igarashi, and Dominik Peters. Fair division of a graph. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, pages 135–141, 2017.

[Caragiannis *et al.*, 2019] 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*, pages 527–545, 2019.

[Chaudhury *et al.*, 2020] Bhaskar Ray Chaudhury, Jugal Garg, and Kurt Mehlhorn. Efx exists for three agents. In *Proceedings of the 21st ACM Conference on Economics and Computation*, pages 1–19, 2020.

[Christodoulou *et al.*, 2023] George Christodoulou, Amos Fiat, Elias Koutsoupias, and Alkmini Sgouritsa. Fair allocation in graphs. In *Proceedings of the 24th ACM Conference on Economics and Computation*, pages 473–488, 2023.

[Dehghani *et al.*, 2018] Sina Dehghani, Alireza Farhadi, Mohammad Taghi Hajiaghayi, and Hadi Yami. Envy-free chore division for an arbitrary number of agents. In *Proceedings of the 29th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 2564–2583, 2018.

[Hosseini *et al.*, 2021] Hadi Hosseini, Sujoy Sikdar, Rohit Vaish, and Lirong Xia. Fair and efficient allocations under lexicographic preferences. In *Proceedings of the 35th AAAI Conference on Artificial Intelligence*, pages 5472–5480, 2021.

[Hummel and Hetland, 2022] Halvard Hummel and Magnus Lie Hetland. Fair allocation of conflicting items. *Autonomous Agents and Multi-Agent Systems*, 36(1):8, 2022.

[Igarashi and Peters, 2019] Ayumi Igarashi and Dominik Peters. Pareto-optimal allocation of indivisible goods with connectivity constraints. In *AAAI*, pages 2045–2052. AAAI Press, 2019.

[Karp *et al.*, 1975] Richard M Karp, Raymond E Miller, and James W Thatcher. Reducibility among combinatorial problems. *Journal of Symbolic Logic*, 40(4), 1975.

[Li *et al.*, 2022] Bo Li, Yingkai Li, and Xiaowei Wu. Almost (weighted) proportional allocations for indivisible chores. In *Proceedings of the ACM Web Conference 2022*, pages 122–131, 2022.

[Liu *et al.*, 2023] Shengxin Liu, Xinhang Lu, Mashbat Suzuki, and Toby Walsh. Mixed fair division: A survey. *CoRR*, abs/2306.09564, 2023.

[Madathil, 2023] Jayakrishnan Madathil. Fair division of a graph into compact bundles. In *Proceedings of the 32nd International Joint Conference on Artificial Intelligence*, pages 2835–2843, 2023.

[Payan *et al.*, 2023] Justin Payan, Rik Sengupta, and Vignesh Viswanathan. Relaxations of envy-freeness over graphs. In *Proceedings of the 2023 International Conference on Autonomous Agents and Multiagent Systems*, pages 2652–2654, 2023.

[Plaut and Roughgarden, 2020] Benjamin Plaut and Tim Roughgarden. Almost envy-freeness with general valuations. *SIAM Journal on Discrete Mathematics*, 34(2):1039–1068, 2020.

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