Extending EFX Allocations to Further Multi-Graph Classes

Umang Bhaskar
Tata Institute of Fundamental Research, Mumbai, India

Yeshwant Pandit
Tata Institute of Fundamental Research, Mumbai, India

Abstract

The existence of EFX allocations is one of the most significant open questions in fair division. Recent work by Christodoulou, Fiat, Koutsoupias, and Sgouritsa ("Fair allocation in graphs," EC 2023) establishes the existence of EFX allocations for graphical valuations, when agents are vertices in a graph, items are edges, and each item has zero value for all agents other than those at its endpoints. Thus, in this setting, each good has non-zero value for at most two agents, and there is at most one good valued by any pair of agents. This marks one of the few cases when an exact and complete EFX allocation is known to exist for more than three agents.

In this work, we partially extend these results to multi-graphs, when each pair of vertices can have more than one edge between them. The existence of EFX allocations in multi-graphs is a natural open question given their existence in simple graphs. We show that EFX allocations exist, and can be computed in polynomial time, for agents with cancelable valuations in the following cases: (i) bipartite multi-graphs, (ii) multi-trees with monotone valuations, and (iii) multi-graphs with girth $(2t-1)$, where t is the chromatic number of the multi-graph. The existence of EFX in cycle multi-graphs follows from (i), (iii), and the known existence of EFX for three agents.

2012 ACM Subject Classification Theory of computation → Algorithmic game theory

Keywords and phrases Fair Division, EFX, Multi-graphs

Digital Object Identifier 10.4230/LIPIcs.FSTTCS.2025.15

Related Version *arXiv version*: `https://arxiv.org/abs/2412.06513`

Funding Both authors acknowledge the support of the Department of Atomic Energy, Government of India, under project no. RTI4001.

1 Introduction

In discrete fair division, we are given a set of indivisible items and a set of agents, with the goal of allocating the items *fairly* among the agents. Various formalisations of fairness have been studied in the literature, of which the most popular is *envy-freeness*, which requires that no agent prefers another's allocation to its own. However, envy-free allocations may not always exist, leading to extensive work on relaxations of envy-freeness. In this paper, we study the existence and computation of allocations that are envy-free up to any item, or EFX, in instances that can be represented as multi-graphs. The question of whether EFX allocations exist in general instances is possibly the most tantalizing open question in the fair allocation of resources, and many recent papers have focused on this question, leading to proofs of its existence in ever-larger classes of instances.

In our work, a fair division instance consists of a set of items that is to be allocated to a set of agents, with each item allocated to a single agent. Each agent has a monotone nondecreasing value function over the set of items. Given an allocation, agent i *envies* agent j if agent i's value for its own allocation is less than its value for the set of items allocated to j. An envy-free allocation is one where no agent envies another. Envy-free allocations do not exist in general, and the closest relaxation then studied is envy-freeness up to any good, or EFX. An EFX allocation allows an agent to envy another, but the removal of *any* item from the envied agent's allocation should eliminate the envy as well.

EFX allocations were first studied by Caragiannis et al. [14], who established its relation to other notions of fairness, but left the question of existence open. If all agents have the same valuation, then an EFX allocation exists [25]. This can be extended to show the existence for two agents with distinct valuations. This was extended to multiple agents with two distinct additive valuations [24], then to three agents when at least one has an MMS-feasible valuation [3, 16], and recently to multiple agents with three distinct additive valuations [21]. A number of papers also study partial EFX allocations, when some items may remain unallocated, e.g., [17, 10], as well as approximately EFX allocations [5, 6, 15, 25].

Another widely studied relaxation of envy-freeness, weaker than EFX, is *envy-freeness up to one item* (EF1), which requires that any envy be resolved by removing *some* item from the envied agent's bundle [13, 23]. Unlike EFX, EF1 allocations are known to exist for both monotone and for some non-monotone valuations [23, 7, 12, 9, 11]. Other fairness notions have also been extensively studied in the literature [4].

Given the difficulties in establishing the existence of EFX allocations, researchers study special cases of structured valuations. EFX allocations are known to exist if agents have lexicographic preferences [20]. Another such class of structured valuations is *graphical valuations*, when the instance can be represented as a graph [18]. Here the agents correspond to the vertices and each item is an edge. The restriction is that each item has nonzero marginal value for only the agents at its end-points, and has zero value for all other agents. Thus if $e = \{i, j\}$, then for every other agent k, and any subset S of items, $v_k(S \cup \{e\}) = v_k(S)$. Further, the graph is simple, and thus there is at most one good valued by every pair of agents. For this case, when agents have monotone valuations over incident edges, Christodoulou et al. [18] show that EFX allocations exist, marking one of the rare cases where EFX allocations are shown to exist for more than three agents.

As mentioned in [18], graphical valuations model real-world scenarios with geographical constraints, where an item is only valued by nearby agents. For example, fishing rights in waters that are adjacent to two nations are such an item, valuable to the two nations but useless to others. Afshinmehr et al. [1] give the example of football matches in various formats; each match must be played at the home of one of the two teams in the match, and hence can be thought of as an item to be allocated to one of the two teams playing.

A natural question with graphical valuations is if an EFX *orientation* exists – an EFX allocation where goods are only allocated to agents at their endpoints. Christodoulou et al. [18] show that an EFX orientation may not exist, and it is NP-hard to determine if a given instance has an EFX orientation. Zeng and Mehta [28] characterize graphs that permit EFX orientations. Zhou et al. [29] study variants of EFX in the graphical setting, where an edge may be valued positively or negatively by its endpoints.

Given the existence of EFX in simple graphs, a natural question is if these results extend to multi-graphs when there are multiple edges between vertices (thus, pairs of agents can value multiple goods). Here a 2/3-approximate EFX allocation exists [5], beating the earlier bound of 0.618 for general additive valuations [6]. EFX allocations also exist in multi-graphs with *symmetrical* additive valuations, i.e., when agents have additive valuations, and both agents at the endpoints of a good have the same value for the good [22]. In independent and concurrent work, Afshinmehr et al. [1, 2] show that exact EFX allocations exist in bipartite multi-graphs and cycle multi-graphs. They also characterize when EFX orientations exist in bipartite graphs, and show it is NP-hard to decide if a given instance admits an EFX orientation. Sgouritsa and Sotiriou [26] also independently present a number of results for multi-graphs when agents have general *monotone* valuations over their incident goods. We discuss these papers in Section 1.2.

1.1 Our Contribution

We study EFX allocations in multi-graphs and show that EFX allocations exist in large and important subclasses of multi-graphs. Our results are algorithmic, and for agents with cancelable valuations (defined in the next section), we show that an EFX allocation in these subclasses can be obtained in polynomial time. A feature of our algorithms is that they are iterative and fairly simple to state, based on the well-known cut-and-choose paradigm. While the algorithms are natural, the analysis is complicated and somewhat subtle.

1. Firstly, we show that for bipartite multi-graphs with cancelable valuations, an EFX allocation can be obtained in polynomial time.
2. Secondly, we show that for multi-trees, EFX allocations exist for *general monotone valuations*. Since multi-trees are also bipartite multi-graphs, for multi-trees with cancelable valuations, a polynomial time algorithm exists from the previous result.
3. Lastly, we generalize the result for bipartite multi-graphs, and show that for multi-graphs with girth at least $(2t-1)$, where t is the chromatic number of the graph, EFX allocations can be obtained in polynomial time for agents with cancelable valuations, given the colour classes.[1] Our algorithm here is a natural extension of the algorithm for bipartite multi-graphs, though the analysis is significantly more complicated.

Note that the chromatic number of a cycle multi-graph is 2 for even cycles and 3 for odd cycles. The existence of EFX allocations in even-length cycle multi-graphs follows from Result 1. The existence of EFX allocations in cycle multi-graphs of length 3 follows from prior work [16]. Result 3 implies the existence of EFX allocations in odd-length cycle multi-graphs of lengths at least 5. Hence, our work also shows the existence of EFX allocations for cycle multi-graphs of all lengths.

1.2 Comparison with Recent Work

Here, we compare our results with recent work on EFX existence in multi-graphs.

As noted earlier, Afshinmehr et al. [1, 2] independently and concurrently show the existence of EFX allocations in bipartite and cycle multi-graphs under additive valuations. Similar to earlier work [18], their algorithm for bipartite multi-graphs operates in two phases: the first constructs an EFX orientation with certain properties, and the second transforms it into a complete allocation using "safe" vertices. While their approach and analysis are somewhat involved, we present a simpler and efficient algorithm with a more straightforward analysis for *cancelable* valuations, which generalize additive valuations. Moreover, our algorithm for bipartite multi-graphs naturally extends to show the existence of EFX allocations in multi-graphs with chromatic number t and girth at least $2t-1$. This extension avoids the challenging case – highlighted in Section 5 of [1, 2] – where there are two envied vertices with the multi-edge between them unallocated, which poses a key difficulty in extending results to general multi-graphs. In contrast, the algorithm, as stated in Afshinmehr et al., does not immediately lend itself to any such natural extension beyond bipartite multi-graphs. Afshinmehr et al. also establish the existence of EFX in cycle multigraphs, which follows as a special case of our result for t-chromatic multigraphs with girth at least $2t-1$.

[1] Note that approximating the chromatic number of a graph within $n^{1-\epsilon}$ for all $\epsilon > 0$ is NP-hard [30].

Another independent and concurrent work by Sgouritsa and Sotiriou [26] proves EFX existence for certain multi-graph classes under *monotone* valuations. They show that EFX allocations exist for: (i) multi-graphs where each vertex has at most $\lceil \frac{n}{4} \rceil - 1$ neighbours (where n is the total number of vertices), and (ii) multi-graphs with girth at least 6. In follow-up work on Arxiv, they extend these results to bipartite multi-graphs [27]. These results extend many of ours to monotone valuations, except for multi-graphs with chromatic number 3 and girth 5 (i.e., $t = 3$), where we establish EFX existence. As in other papers [18, 1, 2], the algorithms by Sgouritsa and Sotiriou follow a multi-phase approach: the first phase computes an initial allocation (orientation) satisfying certain properties, the second aims to reduce the number of envied vertices, and the third assigns the remaining unallocated bundles to non-envied (safe) vertices who value them at zero. Our algorithms offer a different and simpler approach to those proposed in these other papers.

2 Notation and Preliminaries

Multi-graphs. A multi-graph $G = (V, E)$ is a graph with possibly multiple edges between each pair of vertices. A function r maps each edge $e \in E$ to an unordered pair of vertices, which are its endpoints. We use $E_{u,w}$ for the set of parallel edges between the vertices u, w (hence $E_{u,w} = E_{w,u}$). We define $n := |V|$ and $m := |E|$. For a vertex u, N_u is the set of neighbours $\{w \in V : \exists e \in E, r(e) = \{u, w\}\}$.

Given a multi-graph, we can define paths and cycles as in simple graphs without parallel edges. A path $P = (v_1, \ldots, v_k)$ is a sequence of distinct vertices so that each pair of consecutive vertices are neighbours. The length of a path is one less than the number of vertices. A cycle $C = (v_1, \ldots, v_{k-1}, v_k)$ is a sequence of distinct vertices so that each pair of consecutive vertices are neighbours, and v_1, v_k are neighbours.

Fair division. In a fair division instance on a multi-graph $G = (V, E)$, each vertex $u \in V$ corresponds to an agent u, and each edge e corresponds to a good. Throughout this article, we will use the terms agents and vertices interchangeably, and the terms items and edges interchangeably. Each agent u has a valuation function $v_u : 2^E \to \mathbb{Z}_+$ that maps subsets of goods to an integer value. A *bundle* $S \subseteq E$ is simply a subset of goods.

The multi-graph restricts the valuation functions in the following way: Each agent $u \in V$ has zero marginal value for any good not adjacent to it. Thus for a good g if $u \notin r(g)$, then for any subset of goods $S \subseteq E$, $v_u(S \cup \{g\}) = v(S)$. We say that agent u *values* a good g if $u \in r(g)$, i.e., the edge g is adjacent to u, else it does not value the good g. Extending this definition, we say that agent u values a bundle $S \subseteq E$ if S contains some good that u values, else u does not value the bundle S.

An allocation $A = (A_u)_{u \in V}$ is a partition of the set of items (or edges) among the agents. A partial allocation is a partition of a subset of items. An allocation A is *envy-free* (EF) if for all agents u, w, $v_u(A_u) \geq v_u(A_w)$. An allocation A is *envy-free up to any good* (EFX) if for all agents u, w, either $v_u(A_u) \geq v_u(A_w)$, or $v_u(A_u) \geq v_u(A_w \setminus \{x\})$, for all $x \in A_w$. An orientation is an allocation where each item is allocated to an agent that values it (thus if $r(e) = \{u, w\}$, then $e \in A_u \cup A_w$).

Given an allocation A (or a partial allocation), we can define an *envy graph* G_A among the agents: The vertices of G_A are the agents, and there is a directed edge (u, w) if agent u envies agent w, i.e., $v_u(A_u) < v_u(A_w)$. Note that an envy graph is simple.

Given an allocation A and the corresponding envy graph G_A, let $C = (u_1, u_2, \ldots, u_k)$ be a directed cycle in the envy graph. We use the term *resolving the envy cycle C* to refer to the modification that exchanges bundles along the cycle: agent u_k gets the bundle currently allocated to u_1, and each agent u_i gets the bundle currently allocated to u_{i+1}, for $i < k$. The modified allocation is called A^C.

Valuation functions. We will be interested in a few different classes of valuation functions. A valuation function $v : 2^E \to \mathbb{Z}_+$ is *monotone nondecreasing* (or just monotone) if for every $S \subseteq E$ and $g \in E$, $v(S \cup \{g\}) \geq v(S)$.

A monotone valuation function is *cancelable* if for $S, T \subseteq E$ and $g \notin S \cup T$, if $v(S) \geq v(T)$ then $v(S \cup \{g\}) \geq v(T \cup \{g\})$. These functions were defined in prior work on EFX allocations for four agents [10]. We only consider monotone cancelable functions, and hence say cancelable to mean monotone cancelable functions. We in particular need the following two properties of cancelable functions.

▶ **Proposition 1.** *Let v be a cancelable function. Then given disjoint sets of goods S_1, S_2, T_1, T_2, and $v(S_1) \geq v(T_1)$, $v(S_2) \geq v(T_2)$, it follows that $v(S_1 \cup S_2) \geq v(T_1 \cup T_2)$.*

Proof. By cancelability, $v(S_1 \cup S_2) \geq v(T_1 \cup S_2)$ and $v(T_1 \cup S_2) \geq v(T_1 \cup T_2)$, and hence $v(S_1 \cup S_2) \geq v(T_1 \cup T_2)$. ◀

For two agents with monotone valuations, EFX allocations are known to exist [25]. However, even for two agents with submodular valuations, computing an EFX allocation is PLS-complete and requires an exponential number of queries [19]. For cancelable valuations, we can compute an EFX allocation in polynomial time [8, 19].[2]

▶ **Proposition 2.** *For two identical agents with cancelable valuations, an EFX allocation can be computed in polynomial time.*

Cut and Choose. Cut and choose is a standard procedure to obtain an EFX allocation for two agents. Given agents u and w and a set of items S, one agent "cuts" and the other agent "chooses". If agent u "cuts", we assume there are two identical agents with valuation function v_u, and use the algorithm from Proposition 2 to obtain a partition (S^1, S^2) of S that is EFX for identical agents with valuation function v_u. Then agent w "chooses" the bundle with a higher value for it, while agent u is assigned the other bundle. The allocation is clearly envy-free for agent w, and is EFX for agent u. Note that for two agents with cancelable valuations, this gives us a polynomial time algorithm to obtain an EFX allocation.

The cut-and-choose procedure is central to our algorithm. In Section 3, when we deal with bipartite multi-graphs $G = (L \cup R, E)$, for allocating a set of parallel edges $E_{u,w}$ with $u \in L$, $w \in R$, agent $w \in R$ will always cut (thus the agent in the right bipartition always cuts). In Section 5, when we deal with t-chromatic multi-graphs, and partition the vertex set $V = C_1 \cup \ldots \cup C_t$ where vertices in C_i are the same colour, we think of the vertices in C_i as being to the left of C_{i+1}. For a set of parallel edges $E_{u,w}$, as in the bipartite case, the vertex to the *right* will always cut.

We use $\textsc{Cut}(u, S)$ to refer to the use the algorithm from Proposition 2 to obtain a partition (S^1, S^2) of S that is EFX for identical agents with valuation function v_u.

[2] The actual result holds for the more general class of *weakly well-layered* valuations.

3 EFX Allocations in Bipartite Multi-graphs

We first show that for bipartite multi-graphs and agents with cancelable valuations, an EFX allocation always exists, and can be computed in polynomial time. Let $V = L \cup R$ be the bipartition of the vertex set. We first define some notation we will use for this case.

Structures

For each vertex $u \in L$, we define the *structure* St_u as the subgraph induced by $u \cup N_u$. Note that since $N_u \subseteq R$, all edges in St_u are between u and vertices in N_u. We say u is the *root* of the structure. Each iteration in our algorithm will *resolve* the structure St_u, for some $u \in L$, by which we mean that it will assign all the edges in St_u. We also say that a vertex u is resolved to mean that St_u is resolved.

For allocating a set of parallel edges $E_{u,w}$ with $u \in L$, $w \in R$, agent $w \in R$ will always cut (thus the agent in the right bipartition always cuts). For each $w \in N_u$, let $(E^1_{u,w}, E^2_{u,w}) = \textsc{Cut}(w, E_{u,w})$ be the partition of $E_{u,w}$ returned when agent w cuts. Define $S_{u,w} := \arg\max\{v_u(E^1_{u,w}), v_u(E^2_{u,w})\}$ as the bundle preferred by $u \in L$, and $T_{u,w} := \arg\max\{v_w(E^1_{u,w}), v_w(E^2_{u,w})\}$ as the bundle preferred by w. Ties are broken arbitrarily to ensure that $S_{u,w}$ and $T_{u,w}$ are single bundles. If either agent is indifferent between the two bundles, we break ties so that $S_{u,w} \neq T_{u,w}$. Further, we define $\bar{S}_{u,w} = E_{u,w} \setminus S_{u,w}$ and $\bar{T}_{u,w} = E_{u,w} \setminus T_{u,w}$.

For a structure St_u, we further define a *favourite* neighbour f_u as follows:

$$f_u := \arg\max_{w \in N_u} v_u(S_{u,w})\,.$$

We break ties arbitrarily to ensure that f_u is a single vertex. Then f_u is a neighbour who offers u the highest-value bundle after cutting the adjacent edges. The other neighbours are simply called *ordinary* neighbours.

Our algorithm is then simple to describe, and is formally given as Algorithm 1. We consider the vertices in L in turn. For vertex $u \in L$, we consider the structure St_u and resolve it, as follows. Each neighbour $w \in N_u$ cuts the bundle $E_{u,w}$. Each *ordinary* neighbour $w \neq f_u$ gets their higher-value bundle $T_{u,w}$. Let $\mathrm{LO}_u = \bigcup_{w \in N_u \setminus \{f_u\}} \bar{T}_{u,w}$ be the union of the left-over goods from the ordinary neighbours of u. For the favourite neighbour f_u, if $S_{u,f_u} \neq T_{u,f_u}$ – both u and f_u prefer different bundles when f_u cuts E_{u,f_u} – then u gets $S_{u,f_u} \cup \mathrm{LO}_u$, and f_u gets T_{u,f_u}. In this case, everyone gets their largest value bundle, and there is no envy. On the other hand, if $S_{u,f_u} = T_{u,f_u}$ – both u and f_u prefer the same bundle when f_u cuts E_{u,f_u} – then u is offered the choice of S_{u,f_u} or $\bar{S}_{u,f_u} \cup \mathrm{LO}_u$. If u prefers the former, then it gets S_{u,f_u}, and f_u gets $\bar{S}_{u,f_u} \cup \mathrm{LO}_u$. In this case, f_u may envy u, but this is EFX. If u prefers $\bar{S}_{u,f_u} \cup \mathrm{LO}_u$, it gets this bundle, and f_u gets $S_{u,f_u} (= T_{u,f_u})$. Again, in this case, there is no envy.

We note the following properties of the algorithm. In each iteration of the outer for loop, a vertex $u \in L$ is chosen, and the goods in St_u are assigned to the agents in St_u. The allocation to all other agents remains unchanged. Since St_u does not contain any agent from L other than u, an agent $u' \in L$ has no allocated goods until it is resolved, and the allocation to agent u' is not changed after it is resolved. Finally, no goods are removed from an agent once assigned.

▶ **Theorem 3.** *Algorithm 1 runs in polynomial time, and returns an EFX allocation.*

The polynomial running time is easily seen, and is because the procedure $\textsc{Cut}(u, S)$ runs in polynomial time for cancelable valuations. The EFX property follows immediately from the next lemma.

Algorithm 1 Bipartite-EFX.

Require: Bipartite multi-graph $G = (L \cup R, E)$, and vertices with cancelable valuations over E.
Ensure: EFX allocation A.

Initially, for each vertex u, $A_u = \emptyset$
for each vertex $u \in L$ **do**
  for each vertex $w \in N_u$ **do**
    $(E^1_{u,w}, E^2_{u,w}) = \textsc{Cut}(w, E_{u,w})$. ▶ Returns an EFX partition for identical agents
    $S_{u,w} = \arg\max_{S \in \{E^1_{u,w}, E^2_{u,w}\}} v_u(S)$
    $T_{u,w} = \arg\max_{S \in \{E^1_{u,w}, E^2_{u,w}\}} v_w(S)$. ▶ Preferred bundles for u and w from $E_{u,w}$
  $f_u = \arg\max_{w \in N_u} v_u(S_{u,w})$
  $\mathrm{LO}_u = \bigcup_{w \in N_u \setminus \{f_u\}} \bar{T}_{u,w}$ ▶ Left-over bundles from all ordinary neigbours
  $A_w = A_w \cup T_{u,w}$ for each vertex $w \in N_u \setminus \{f_u\}$ ▶ Ordinary neighbours get their preferred bundle
  if $(S_{u,f_u} = T_{u,f_u})$ **then** ▶ Both u and f_u prefer the same bundle in E_{u,f_u}
    if $(v_u(S_{u,f_u}) > v_u(\bar{S}_{u,f_u} \cup \mathrm{LO}_u))$ **then** ▶ u prefers S_{u,f_u} over all the left-over bundles
      $A_u = S_{u,f_u}$
      $A_{f_u} = A_{f_u} \cup \bar{S}_{u,f_u} \cup \mathrm{LO}_u$
    else ▶ u prefers the left-over bundles
      $A_u = \bar{S}_{u,f_u} \cup \mathrm{LO}_u$
      $A_{f_u} = A_{f_u} \cup S_{u,f_u}$
  else ▶ u and f_u prefer different bundles
    $A_u = S_{u,f_u} \cup \mathrm{LO}_u$
    $A_{f_u} = A_{f_u} \cup T_{u,f_u}$
return A

▶ **Lemma 4.** *Fix an agent $u \in L$. Let A be the partial allocation after resolving St_u. Then the only possible envy in allocation A is that a resolved agent $u' \in L$ is envied by its favourite agent $f_{u'}$. Further, the partial allocation A is EFX.*

Proof. The proof is by induction on the iterations of the outer for loop. The claim is clearly true for the initial empty allocation. Let $\hat{A}$ be the allocation before u is resolved. Then by the induction hypothesis, since u is not yet resolved in $\hat{A}$, no agents in St_u are envied.

In the current iteration, when St_u is resolved, only agents in St_u get goods, while agents not in St_u retain their allocation. Hence, any new envy edges must be to agents in St_u. Further since only agents in St_u value the goods in St_u, any new envy edges must also be from agents in St_u.

Suppose agent w envies agent $w' \notin \mathrm{St}_u$ in $\hat{A}$ (as established, agents in St_u are not envied in $\hat{A}$). Thus $v_w(\hat{A}_w) < v_w(\hat{A}_{w'})$. In the current iteration, the allocation to w' does not change, while w does not lose any goods. Hence any such envy remains EFX. Thus to prove the lemma, we only need to show that any new envy between agents in St_u is from f_u to u, and is EFX.

Consider first an ordinary neighbour w. Of the goods allocated in this iteration, w only values $E_{u,w}$. Of this set, it gets its preferred partition $T_{u,w}$. Then since w did not envy anyone in St_u earlier, for any agent $z \in \mathrm{St}_u$, $v_w(\hat{A}_w) \geq v_w(\hat{A}_z)$. Then

$$v_w(A_w) = v_w(\hat{A}_w \cup T_{u,w}) \geq v_w(\hat{A}_z \cup \bar{T}_{u,w}) \geq v_w(A_z)\,.$$

The first inequality is because valuations are cancelable, and w prefers $\hat{A}_w$ to $\hat{A}_z$, and $T_{u,w}$ to $\bar{T}_{u,w}$. The second inequality is because in this iteration, any agent other than w either does not receive any good valued by w, or receives the bundle $\bar{T}_{u,w}$.

Consider f_u, agent u's favourite neighbour. Again of the goods allocated in this iteration, f_u is only interested in E_{u,f_u}. Of this set, it either gets its preferred bundle T_{u,f_u} (in either Line 14 or in Line 17), or it gets the bundle $\bar{S}_{u,f_u}$ in Line 11. In the former case, since the valuations are cancelable and agent f_u did not earlier envy any agent in St_u, agent f_u does not envy any agent in St_u in allocation A either. In the latter case, f_u gets the bundle $\bar{S}_{u,f_u}$, and hence may envy u since agent u gets $S_{u,f_u} = T_{u,f_u}$. But note that then $A_u = S_{u,f_u}$, and this is an EFX partition since f_u cuts the set of items E_{u,f_u}. Thus, the only possible new envy edge from f_u is to u, and this is EFX.

Lastly, consider agent u. We show that u will not envy any agent in St_u. If u gets $S_{u,f_u} \cup \mathrm{LO}_u$ (Line 17), then by definition S_{u,f_u} is preferred over all the other bundles allocated to the other agents in St_u, and u additionally gets all the left-over bundles. Otherwise, u gets a choice between S_{u,f_u} and $\bar{S}_{u,f_u} \cup \mathrm{LO}_u$; the other bundle is given to f_u. Then clearly u does not envy f_u. The bundle S_{u,f_u} has a higher value for u than $T_{u,w}$ for any neighbour w. Since u chooses the bundle with the highest value, u does not envy any ordinary neighbour either. ◀

4 EFX Allocations for Monotone Valuations in Tree Multi-graphs

We now show that for monotone valuations, an EFX allocation exists in tree multi-graphs. Note that even for two agents with monotone submodular valuations, computing an EFX allocation is PLS-complete [19], and hence a polynomial-time algorithm for tree multi-graphs may not exist. Since trees are also bipartite, a polynomial time algorithm for *cancelable* valuations follows from the previous section.

Our algorithm for trees is recursive and again utilizes the procedure $\textsc{Cut}(v, S)$, that obtains an EFX partition (S^1, S^2) of S for two identical agents with valuation v. Let $T = (V, E)$ be the tree multi-graph, and ℓ be a leaf with parent p. Then $E_{\ell,p}$ are the edges between ℓ and p, and no other agent values these goods. Inductively, let A' be an EFX allocation of goods $E \setminus E_{\ell,p}$ to agents $V \setminus \{\ell\}$. Our objective is then to extend A' to an EFX allocation of goods E.

For this, let (S^1, S^2) be the partition obtained when p cuts $E_{\ell,p}$. We define $A_\ell = \arg\max_{S \in \{S^1, S^2\}} v_\ell(S)$ and $\bar{A}_\ell = E_{\ell,p} \setminus A_\ell$. Note that ties are broken arbitrarily, ensuring that A_ℓ is a single bundle. Agent ℓ gets its preferred bundle A_ℓ. If p is not envied in allocation A', then it gets $\bar{A}_\ell$, and the algorithm terminates. If p is already envied in allocation A', it cannot get additional goods. Let s be a source in the envy graph for A' with a path to p. We add $\bar{A}_\ell$ to agent s's allocation. If p does not envy s now, the algorithm terminates. Else, there is an envy cycle where p envies s. Resolving the envy cycle gives us an EFX allocation.

▶ **Theorem 5.** *Given a tree multi-graph, $T = (V, E)$ with monotone valuations, Algorithm Tree-EFX returns an EFX allocation A.*

Proof. By induction, assume that A' is an EFX allocation on $T' = (V \setminus \{\ell\}, E \setminus E_{\ell,p})$. Note that given an EFX allocation, resolving envy cycles does not destroy this property, since each agent's value does not decrease, and bundles are merely reassigned. Hence, the partial allocation in Line 11 is EFX for agents $V \setminus \{\ell\}$, and $A_\ell = \emptyset$.

On allocating A_ℓ to agent ℓ, clearly ℓ does not envy any agent. The only agent that could envy ℓ is agent p, since only agents p and ℓ value the bundle A_ℓ. Further, since p gets to cut the bundle $E_{\ell,p}$, as long as p possesses a bundle with value at least $v_p(\bar{A}_\ell)$, its envy towards ℓ will be EFX.

Algorithm 2 Tree-EFX.

Require: A tree multi-graph $T = (V, E)$ with $|V| \geq 2$ and monotone valuations.
Ensure: An EFX allocation A

1: **if** $(|V| = 2)$ **then**
2: Let $V = \{\ell, p\}$
3: $(S^1, S^2) = \textsc{Cut}(p, E_{\ell,p})$
4: $A_\ell = \arg\max_{S \in \{S^1, S^2\}} v_\ell(S)$, $A_p = \bar{A}_\ell$
5: **return** $A = (A_\ell, A_p)$
6: Let ℓ be a leaf in T
7: $A' =$ Tree-EFX$(V \setminus \{\ell\}, E \setminus E_{\ell,p})$ ▶ Recursive call on the smaller tree without leaf ℓ
8: $G_{A'}$ is the envy graph for allocation A'
9: **while** (there is an envy-cycle C in $G_{A'}$) **do** ▶ Resolve any envy-cycles
10: $A' = A'^C$
11: $A_u = A'_u$ for all vertices $u \neq \ell$
12: $(S^1, S^2) = \textsc{Cut}(p, E_{\ell,p})$ ▶ p cuts the bundle $E_{\ell,p}$
13: $A_\ell = \arg\max_{S \in \{S^1, S^2\}} v_\ell(S)$ ▶ Leaf ℓ gets its preferred bundle
14: **if** (p is a source in $G_{A'}$) **then**
15: $A_p = A_p \cup \bar{A}_\ell$ ▶ If p is not envied, p gets the left-over items
16: **else**
17: Let s be a source with a path P to p. ▶ Else, assign left-over items to s, resolve any envy cycle.
18: $A_s = A_s \cup \bar{A}_\ell$
19: **if** $(v_p(A_p) < v_p(A_s))$ **then**
20: Let C be the envy-cycle consisting of the edge (p, s) and the s-p path P.
21: $A = A^C$
22: **return** A

If p is a source in the envy graph, then the condition in Line 14 is satisfied and gets the bundle $\bar{A}_\ell$. No agent will envy p, since agents $V \setminus \{p, \ell\}$ do not value the bundle $\bar{A}_\ell$, and ℓ already gets its preferred bundle. Thus the allocation A is EFX.

If p is not a source, then since the envy-graph is acyclic (and ℓ does not envy any agent), there is a source s with a path to p in the envy graph. Let P be this path. We assign $\bar{A}_\ell$ to agent s. The only agent that could now envy s is p. If p envies s, there is an envy cycle, which we resolve. We claim that the resulting allocation is EFX. This is because agent ℓ continues to not envy any agent. Agent p's value for its bundle is now at least $v_p(\bar{A}_\ell)$, and hence any envy towards agent ℓ is EFX. Further agent p's value for its bundle is also at least its earlier value; since resolving the cycle only shifted bundles around, any envy towards agents $V \setminus \{\ell\}$ is also EFX by induction. Finally, for other agents, their value does not decrease, and any envy is unaffected by the shifting of bundles. Hence the allocation is EFX. ◀

5 EFX Allocations in t-Chromatic Multi-graphs with Girth at Least $(2t - 1)$

We now extend our algorithm and analysis to show that EFX allocations exist for agents with cancelable valuations in multi-graphs with girth at least $2t - 1$, where t is the chromatic number of the graph. A multi-graph is t-chromatic if t is the minimum number of colours required to colour the vertices, so no edge has both end-points the same colour. The girth of a multi-graph is the length of the shortest simple cycle, without using parallel edges. For example, bipartite multi-graphs are 2-chromatic with girth at least 4, while the Petersen graph (with parallel edges) is 3-chromatic with girth 5. We will assume that we are given a t-colouring of the multi-graph.

Our algorithm proceeds as follows. Given a multi-graph $G = (V, E)$ coloured with t colours, let $c(u) \in [t]$ be the colour of each vertex $u \in V$. Let $C^i = \{u : c(u) = i\}$ be the independent set consisting of all vertices coloured i. We think of the vertices as ordered from left to right by their colour, and say a vertex u is *to the left of* w if $c(u) < c(w)$. For $i = 1, \ldots, t-1$, define $L^i := C^i$, and $R^i := C^{i+1} \cup \ldots \cup C^t$.

We proceed in phases. In phase i, we consider the bipartite multi-graph $G^i = (V^i = L^i \cup R^i, E^i)$ where an edge $e \in E$ is in E^i if $|r(e) \cap L^i| = |r(e) \cap R^i| = 1$, i.e., each edge is from a vertex in L^i to a vertex in R^i. Note that each vertex $u \in V \setminus C^t$ appears in a left bipartition in exactly one phase, when we consider all edges to vertices to the right of u. Thus each edge appears in exactly one phase.

Now in phase i, we run Algorithm 1 on the bipartite multi-graph G^i. The only change we make is that in Algorithm 1, initially all vertices – in particular, vertices $u \in L$ – had empty allocations. This may not be the case in the current algorithm in the second phase onwards. Hence, we remove Line 1 from Algorithm 1, which initializes the allocation, within the outer for loop. Further in Lines 16 and 19 in Algorithm 3, agent $u \in L^i$ retains its earlier allocation. In Line 13, when $u \in L^i$ is possibly envied by f_u, its earlier allocation passes to f_u. Algorithm 3 formally states the algorithm.

A *phase* is an iteration of the outer loop. In phase i, the vertices in C^i are picked sequentially in the inner for loop.

Structures

For our analysis, we slightly need to redefine *structures*. For a vertex $u \in C^i$, we now define the structure St_u as the subgraph induced by $u \cup (N_u \cap R^i)$. That is, a structure now consists of the subgraph induced by u and all neighbours *to its right*. As before, u is the *root* of the structure. Now each iteration in phase i of our algorithm resolves the structure St_u for some $u \in C^i$, and thus assigns all the edges in St_u. A vertex u is resolved if St_u is resolved.

Fix agent $u \in C^i$. As before, for each $w \in (N_u \cap R^i)$, $(E^1_{u,w}, E^2_{u,w}) = \textsc{Cut}(w, E_{u,w})$ is the partition of $E_{u,w}$ returned when agent w cuts. Define $S_{u,w} := \arg\max\{v_u(E^1_{u,w}), v_u(E^2_{u,w})\}$ as the bundle preferred by u, and $T_{u,w} := \arg\max\{v_w(E^1_{u,w}), v_w(E^2_{u,w})\}$ as the bundle preferred by w. As earlier, ties are broken arbitrarily to ensure that $S_{u,w}$ and $T_{u,w}$ are single bundles. If either agent is indifferent between the two bundles, we break ties so that $S_{u,w} \neq T_{u,w}$. Further, for $S \subseteq E_{u,w}$, we define $\bar{S} = E_{u,w} \setminus S$. Then u's *favourite* neighbour $f_u := \arg\max_{w \in N_u \cap R_i} v_u(S_{u,w})$, with ties broken arbitrarily.
This is the right neighbour that offers u the highest-value bundle after cutting the adjacent edges. The other right neighbours are simply called *ordinary* neighbours.

We note the following properties of our algorithm. In phase i of the algorithm, the inner for loop picks an agent $u \in C^i$, and resolves St_u. The goods in St_u are assigned to the agents in St_u. The allocation to other agents is unchanged. Hence in particular, the allocation to a resolved agent does not change subsequently. Furthermore, while St_u is resolved, agents in St_u do not lose any goods, except possibly u itself. Agent u may however give its entire set of goods to its favourite neighbour f_u (Line 13). However, in this case, agent u gets S_{u,f_u}, which it prefers over its earlier allocation and all the left-over goods. Thus, the value of each agent for its bundle is nondecreasing.

Algorithm 3 t-Chromatic-EFX.

Require: t-Chromatic multi-graph $G = (L \cup R, E)$ with girth $(2t-1)$ and cancelable valuations. C^i is the set of vertices coloured $i \in [t]$.
Ensure: EFX allocation A.

1: Initially, for each vertex u, $A_u = \emptyset$. Resolved agents $\text{Res} = \emptyset$.
2: **for** $i = 1$ to $t-1$ **do**
3: $L^i = C^i$, $R^i = C^{i+1} \cup \ldots \cup C^t$. ▶ Only consider edges between L^i, R^i
4: **for** each vertex $u \in L^i$ **do**
5: **for** each vertex $w \in N_u \cap R^i$ **do**
6: $(E^1_{u,w}, E^2_{u,w}) = \textsc{Cut}(w, E_{u,w})$. ▶ Right vertex cuts
7: $S_{u,w} = \arg\max_{S \in \{E^1_{u,w}, E^2_{u,w}\}} v_u(S)$
8: $T_{u,w} = \arg\max_{S \in \{E^1_{u,w}, E^2_{u,w}\}} v_w(S)$.
9: $f_u = \arg\max_{w \in N_u \cap R^i} v_u(S_{u,w})$
10: $\text{LO}_u = \bigcup_{w \in N_u \setminus \{f_u\} \cap R^i} \bar{T}_{u,w}$ ▶ Left-over bundles from all ordinary neigbours
11: $A_w = A_w \cup T_{u,w}$ for all $w \in N_u \setminus \{f_u\} \cap R^i$ ▶ Ordinary neighbours get their preferred bundle
12: **if** $(S_{u,f_u} = T_{u,f_u})$ **then** ▶ Both u and f_u prefer the same bundle in E_{u,f_u}
13: **if** $(v_u(S_{u,f_u}) > v_u(A_u \cup \bar{S}_{u,f_u} \cup \text{LO}_u))$ **then** ▶ u prefers S_{u,f_u} over left-over and its own goods
14: $A_{f_u} = A_{f_u} \cup A_u \cup \bar{S}_{u,f_u} \cup \text{LO}_u$
15: $A_u = S_{u,f_u}$
16: **else** ▶ u prefers the left-over bundles
17: $A_u = A_u \cup \bar{S}_{u,f_u} \cup \text{LO}_u$
18: $A_{f_u} = A_{f_u} \cup S_{u,f_u}$
19: **else** ▶ u and f_u prefer different bundles
20: $A_u = A_u \cup S_{u,f_u} \cup \text{LO}_u$
21: $A_{f_u} = A_{f_u} \cup T_{u,f_u}$
22: $\text{Res} = \text{Res} \cup \{u\}$ ▶ Bookkeeping: u is resolved.
23: **return** A

Much of our effort is towards proving Lemma 11, which shows that any envy is within a structure, and is from a favourite neighbour f_u to the root u. In particular, we want to avoid the following. Say for some interim allocation A, agent u' values goods in both A_u and A_{f_u}. Then if A_u is transferred to f_u in Line 13, agent u' should not envy f_u.

We show this by contradiction: that if agent u' does indeed start envying f_u, then it must have short paths to both u and f_u, and hence, taken with any edge between u and f_u, there is a cycle of length less than $2t-1$. We do this in two steps: Claim 7 shows that if u' envies the bundle of goods $A_u \cup A_{f_u}$, then these must contain a good from $\text{St}_{u'}$. Claims 9 and 10 then establish bounds on the distance from u' for u and f_u.

Next, we state the claims required to prove Lemma 11. Our first claim is that an unresolved agent does not envy the *entire* set of goods held by all other unresolved vertices.

▷ Claim 6. Let A and Res be the allocation and the set of resolved vertices at Line 5. Let z be an unresolved agent. Then

$$v_z(A_z) \geq v_z\left(\bigcup_{w \notin \text{Res},\, w \neq z} A_w\right).$$

Proof. Fix an unresolved agent z. We prove this by induction on the iterations of the inner loop. Initially, the allocation is empty, and the claim is trivially satisfied. Assume this is true prior to the loop when a vertex u is resolved. Let $\hat{A}$, $\widehat{\mathrm{Res}}$ be the allocation and set of resolved vertices before u is resolved, and A, $\mathrm{Res} = \widehat{\mathrm{Res}} \cup \{u\}$ be the allocation and set of resolved vertices after. Note that the allocation to resolved agents does not change subsequently. Hence if z now envies the set of goods held by unresolved agents, new goods that z values must be allocated to unresolved agents.

In this loop, only the allocation to agents in St_u is modified. Further, none of the agents in St_u are resolved prior to the loop, i.e., $\mathrm{St}_u \cap \widehat{\mathrm{Res}} = \emptyset$. Hence $v_z(\hat{A}_z) \geq v_z\left(\cup_{u' \in \mathrm{St}_u,\, u' \neq z} \hat{A}_{u'}\right)$ by the induction hypothesis. Now if $z \notin \mathrm{St}_u$, then no additional goods it values are allocated in this loop. Hence the claim holds by induction. Thus, since $z \neq u$ (since z is unresolved after the loop), it must be that z is a neighbour to the right of u. Of the goods allocated in this loop, z only values $E_{u,z}$.

If z is an ordinary neighbour, then z additionally gets its preferred bundle $T_{u,z}$, while the bundle $\bar{T}_{u,z}$ is allocated to other agents. Hence, since valuations are cancelable, the claim remains true. On the other hand, if $z = f_u$, then z could lose its preferred bundle $T_{u,z}$ to agent u. But in this case agent u is resolved, hence $u \notin \mathrm{Res}$. Since z additionally gets $\bar{T}_{u,z}$ and does not lose any goods, the claim remains true after the loop also. ◁

The next claim builds on the previous claim to show that if any vertex z envies the union of goods held by some set S of unresolved agents, then z must be resolved, and some agent in S must hold a good from St_z.

▷ Claim 7. Let A and Res be the allocation and the set of resolved vertices at Line 5. Let z be any agent, and S is a subset of unresolved vertices so that

$$v_z(A_z) < v_z\left(\bigcup_{w \in S, z \notin S} A_w\right).$$

Then z is resolved, and for some agent $w \in S$, A_w contains some good $g \in \mathrm{St}_z$.

Proof. By Claim 6, if z is unresolved, then $v_z(A_z) \geq v_z\left(\cup_{w \in S,\, w \neq z} A_w\right)$. Hence z must in fact be resolved. We prove the contrapositive by induction, that if a subset of unresolved agents S do not hold any good from St_z, then $v_z(A_z) \geq v_z(\cup_{w \in S} A_w)$. Clearly, the claim holds in any iteration before z is resolved.

Consider the iteration where z is resolved. Every agent in St_z receives goods from St_z, hence, we only consider subsets S consisting of agents not in St_z. But the allocation for these agents does not change, and z's value for its bundle is nondecreasing, hence the claim holds in this iteration.

Consider any later iteration, where an agent $u \neq z$ is resolved. Then $u \notin S$, since u is resolved. Let $\hat{A}$ be the allocation before resolving St_u, and A the allocation after. Again, only the allocation for agents in St_u changes, hence by induction we only need to consider subsets S that contain agents in St_u. Note that since agent z is already resolved, z does not value any goods in St_u. Hence the allocation of goods in St_u does not affect the right hand side of the inequality in the claim.

Other than the allocation of goods in St_u, the only change that happens in the algorithm is that the prior bundle $\hat{A}_u$ may be transferred to f_u. Thus if $\hat{A}_u$ is not transferred to f_u, or $f_u \notin S$, then z's value for the aggregate bundle $\cup_{w \in S} A_w$ is unchanged, and the claim holds.

Thus, $f_u \in S$, $u \notin S$, and the only relevant change that occurs in the iteration is that the bundle $\hat{A}_u$ is transferred to agent f_u. Note that since $f_u \in S$, f_u does not hold any goods from St_z, and hence $\hat{A}_u$ does not contain any goods from St_z. Now in the allocation $\hat{A}$, consider the subset $\hat{S} = S \cup \{u\}$. Then

$$\begin{aligned} v_z(A_z) = v_z(\hat{A}_z) &\geq v_z\left(\bigcup_{w \in \hat{S}} \hat{A}_w\right) \quad \text{(by the induction hypothesis)} \\ &= v_z\left(\hat{A}_u \cup \bigcup_{w \in S} \hat{A}_w\right) = v_z\left(\bigcup_{w \in S} A_w\right), \end{aligned}$$

proving the claim. ◁

▶ **Proposition 8.** *For an agent u, let $\hat{A}$ and A be the allocations before and after St_u is resolved. If for some agent z and good g, $g \in \hat{A}_z$ but $g \notin A_z$, then $z = u$ and $g \in A_{f_u}$. Thus in a phase, any good allocated before the phase began can only be transferred from the root of a structure to its favourite neighbour.*

Proof. As noted, when St_u is resolved, the goods in the structure are allocated to the agents in the structure. If Line 13 is executed, then additionally the bundle A_u is transferred to agent f_u (and agent u gets the bundle S_{u,f_u}). Thus the only way that a good that is assigned to an agent before St_u is resolved can be moved, is if it was assigned to u and then is transferred to agent f_u. The proposition follows.

Since a root only appears once in a phase, a good once transferred in a phase cannot be transferred again, and hence any good allocated prior to a phase can only be transferred once, from the root of a structure to its favourite neighbour. ◀

The next few results show that a good valued by an agent z cannot be very far from z.

▷ Claim 9. Let A be the allocation at Line 5. If agent z values a good g and $g \in A_w$, then $\mathrm{dist}(z, w) \leq c(w)$ along allocated edges.

Proof. By Proposition 8, a good can only be transferred rightward, from an agent being resolved to its favourite neighbour. If $c(w) = 1$, then once St_w is resolved, w's allocation does not change. Thus if $g \in A_w$ and $c(w) = 1$, w received this good in phase 1 when St_w was resolved. Since z values g, z must be in N_w, and there is a path of length 1 between z and w along the edge set $E_{w,z}$ satisfying the claim.

Now suppose $c(w) \geq 2$. Consider the phase when g is initially allocated, say to an agent w' when St_u is resolved. Since z values g, $z \in \mathrm{St}_u$. Hence z and w' have a path of length 2 along the edges of St_u. From Proposition 8, in each subsequent phase, g is transferred from the root of a structure to its favourite neighbour. Suppose g is allocated to w in phase k. Then w must be a right neighbour for the agent being resolved, and hence the colour $c(w) \geq k + 1$. There are at most $k - 1$ phases after the phase g is initially allocated. Thus there is a path of length $k - 1 + 2 \leq c(w)$ along allocated edges from w to z. ◁

If z not only values g, but $g \in \mathrm{St}_z$ and is held by agent w, then we can get a tighter bound on $\mathrm{dist}(z, w)$.

▷ Claim 10. Let A be the allocation at Line 5. Let there be agents z, w, and good $g \in \mathrm{St}_z \cap A_w$. Then $\mathrm{dist}(z, w) \leq c(w) - c(z)$ along allocated edges.

Proof. Since $g \in \mathrm{St}_u$, when St_u is resolved, it is assigned either to u or to a neighbour to the right. Further any reassignment of g transfers it to a neighbour further to the right. Hence if $u \neq w$, then w must be to the right of u, hence $c(w) > c(u)$.

Now when g is initially allocated in phase $c(u)$, it is allocated to an agent w' at a distance 1 from u along the edges $E_{u,w'}$. Suppose g is allocated to agent w in phase k. Then $c(w) > k$, and there are at most $k - c(u)$ phases after the phase when g is initially allocated. Thus there is a path of length $k - c(u) + 1 \leq c(w) - c(u)$ along allocated edges from u to w. ◁

We now prove our main lemma.

▶ **Lemma 11.** *Let A be the allocation at Line 5. Then if w envies u, then u is a resolved vertex, and $w = f_u$ (i.e., w is u's favourite vertex in St_u).*

Proof. We will prove the lemma by induction. For the base case, the initial allocation is empty and trivially satisfies the two properties. Now fix any vertex u. Let $\hat{A}$ be the allocation before the structure rooted at u is resolved, and A be the allocation after resolving St_u.

None of the vertices in St_u are resolved in $\hat{A}$, and hence by the induction hypothesis, none of these are envied in $\hat{A}$. When St_u is resolved, only the allocation to agents in St_u is modified, while the allocation to other agents is unchanged. The value of agents in St_u does not decrease. Hence, any new envy edge must be towards agents in St_u.

Consider an ordinary neighbour w in St_u. Agent w is not envied in $\hat{A}$. It receives its preferred bundle $T_{u,w}$; this is only valued by u and w, hence no agent other than u will envy w. To see that u also will not envy w, we first claim that $v_u(\hat{A}_w) = 0$, i.e., u does not value w's bundle in $\hat{A}$. To prove this, assume for a contradiction that u values w's bundle. Then by Claim 9, there exists a u-w path along allocated edges of length at most t. Including the (unallocated) edges $E_{u,w}$, this gives a cycle of length $t+1$. If $t = 2$, this gives us a 3-cycle, which would mean that the graph is not 2-colourable. If $t > 2$, then $t + 1 < 2t - 1$, but the graph has girth at least $2t - 1$. In either case we get a contradiction. Thus, $v_u(\hat{A}_w) = 0$.

Further, $v_u(A_u) \geq v_u(T_{u,w})$. This is because agent u prefers S_{u,f_u} over $T_{u,w}$, and always has the choice of keeping S_{u,f_u}. Hence, it ends up with a bundle of value at least that of $T_{u,w}$. Then since $v_u(\emptyset) \geq v_u(\hat{A}_w)$ and $v_u(A_u) \geq v_u(T_{u,w})$, since valuations are cancelable, $v_u(A_u) \geq v_u(A_w)$. Thus, an ordinary neighbour $w \in \mathrm{St}_u$ is not envied.

Now consider agent u. As shown, agent u is not envied in $\hat{A}$. While resolving St_u, the only goods agent u can possibly receive are goods in St_u, and hence no agent outside St_u will envy u. Within St_u, all ordinary neighbours receive their preferred bundle, and hence will not envy u. Thus the only agent that could possibly envy u after resolving St_u is f_u.

Now if f_u gets its preferred bundle T_{u,f_u} (in Lines 16 and 19), it does not envy u. If agent f_u gets $\bar{T}_{u,f_u}$ (in Line 13), then this is an EFX partition for f_u (since f_u cut sE_{u,f_u}), and agent u only gets T_{u,f_u}. In this case, f_u envies u, but the envy is EFX.

Lastly, consider agent f_u. This case is more complicated than the earlier cases, since the bundle $\hat{A}_u$ previously held by u could be transferred to f_u in Line 13. We will show however that no agent envies f_u.

First, let's consider agents in St_u. Again, the ordinary neighbours w did not envy f_u earlier, and from the goods $E_{u,w}$ they get their preferred bundle $T_{u,w}$. Hence, since valuations are cancelable, they will not envy f_u.

Agent u, as argued previously, by Claim 9, does not value $\hat{A}_{f_u}$, else there is a cycle of length strictly smaller than $2t - 1$. Further agent u gets a bundle of value at least that of S_{u,f_u}, and hence will not envy f_u.

For the last case, consider any agent $u' \notin \mathrm{St}_u$. In allocation $\hat{A}$, u' does not envy u or f_u. Since u' does not value goods in E_{u,f_u}, if it envies A_{f_u}, clearly (i) it must value both $\hat{A}_u$ and $\hat{A}_{f_u}$, and (ii) it must envy $\hat{A}_u \cup \hat{A}_{f_u}$.

Suppose $c(u') \geq 2$. Then from Claim 7, either u or f_u holds a good from $\mathrm{St}_{u'}$ (say u). By Claim 10, u is then at distance $c(u) - c(u')$ from u' along allocated edges. From (i) in the previous paragraph, and Claim 9, f_u is at distance at most $c(f_u)$ from u' along allocated edges. Thus, there is u-f_u path along allocated edges (via u') of length $c(f_u) + c(u) - c(u')$ $\leq 2t - 3$ (since $c(u) \leq t - 1$ and $c(u') \geq 2$). But then including the (unallocated) edges E_{u,f_u}, there is a cycle of length $2t - 2$, which is a contradiction, since we assume the graph has girth at least $2t - 1$.

Finally, suppose $c(u') = 1$. Then note that all the goods that u' values are in $\mathrm{St}_{u'}$. Since both u and f_u possess goods that u' values, by Claim 10, they are at distance $c(u) - 1$ and $c(f_u) - 1$ respectively from u' along allocated edges. Since $c(u) \leq t - 1$, there is a path of length $2t - 3$ from u to f_u (via u') along allocated edges. But then including the (unallocated) edges E_{u,f_u}, there is a cycle of length $2t - 2$, which is again a contradiction. Thus, agent f_u is not envied in the allocation A. This completes the proof. ◀

We now can easily prove that the allocation obtained is EFX.

▶ **Theorem 12.** *Algorithm 3 obtains an EFX allocation for agents with cancelable valuations in polynomial time.*

Proof. The proof for the running time for Algorithm 3 is obtained easily, since for cancelable valuations $\textsc{Cut}(u, S)$ runs in polynomial time, and the other steps are straightforward.

We now show that the algorithm obtains an EFX allocation. As before, the proof is by induction on the iterations of the inner for loop. For the base case, the statement clearly holds for the empty allocation. Then consider an agent u, and let $\hat{A}$ and A be the allocations before and after St_u is resolved.

Consider first an envy edge from agent w to agent u' in allocation $\hat{A}$. We will show that this envy remains EFX in allocation A as well. By Lemma 11, agent u' must be resolved. No agent in St_u is resolved in $\hat{A}$, hence $u' \notin \mathrm{St}_u$. Since only agents in St_u have their allocations modified, agent u''s allocation is not modified. Agent w is either also not in St_u (in which case its allocation also does not change), or is in St_u (in which case its value does not decrease). In either case, either w does not envy u' in A, or the envy remains EFX.

By Lemma 11, the only envy edge that may be in A but not in A' is from f_u to u. This happens in Line 11, when u gets S_{u,f_u}. But then agent f_u gets $\bar{S}_{u,f_u}$. Since agent f_u cuts the bundle E_{u,f_u}, and agent u has no goods besides S_{u,f_u}, the envy is EFX. ◀

Finally, we show that EFX allocations exist in cycle multi-graphs, from the earlier results.

▶ **Theorem 13.** *For agents with cancelable valuations in a cycle multi-graph, EFX allocations exist.*

Proof. We consider three separate cases, depending on the length of the cycle. For even-length cycles, the result follows from Theorem 3, since such a cycle is a bipartite multi-graph. If the length is 3, then the result follows from EFX existence for three agents [16]. Finally, if the cycle has odd length of at least 5, the chromatic number is 3, and the existence follows from Theorem 12, setting $t = 3$, since the girth is least 5. ◀

6 Conclusion

Our paper extends the work on EFX existence to important classes of multi-graphs. We view our results as a significant extension over the work by Christodoulou et al. [18] since their results are for simple graphs. Our work also shows that the well-known cut-and-choose paradigm can be leveraged to obtain results in fairly general settings. Our algorithm is another tool in the toolbox for EFX allocations, different, for example, from the multiple-phase algorithms used in many other papers.

We note that if edges are chores instead of good, i.e., an edge is negatively-valued by both adjacent vertices, an envy-free allocation can be obtained by assigning each edge to a vertex other than its endpoints. The existence of EFX allocations for general multi-graphs with positively valued edges remains an open problem. We also find it interesting that the case of a cycle multi-graph with 3 agents does not yet have a simpler algorithm than for general 3 agents. Coming up with a simpler algorithm here may be crucial to extending our work to general multi-graphs.

References

1 Mahyar Afshinmehr, Alireza Danaei, Mehrafarin Kazemi, Kurt Mehlhorn, and Nidhi Rathi. EFX allocations and orientations on bipartite multi-graphs: A complete picture. *CoRR*, abs/2410.17002, 2024. doi:10.48550/arXiv.2410.17002.

2 Mahyar Afshinmehr, Alireza Danaei, Mehrafarin Kazemi, Kurt Mehlhorn, and Nidhi Rathi. EFX allocations and orientations on bipartite multi-graphs: A complete picture. In *Proceedings of the 24th International Conference on Autonomous Agents and Multiagent Systems*, pages 32–40, 2025. doi:10.5555/3709347.3743514.

3 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 2023, London, United Kingdom, July 9-12, 2023*, page 61. ACM, 2023. doi:10.1145/3580507.3597799.

4 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. doi:10.1016/J.ARTINT.2023.103965.

5 Georgios Amanatidis, Aris Filos-Ratsikas, and Alkmini Sgouritsa. Pushing the frontier on approximate EFX allocations. In Dirk Bergemann, Robert Kleinberg, and Daniela Sabán, editors, *Proceedings of the 25th ACM Conference on Economics and Computation, EC 2024, New Haven, CT, USA, July 8-11, 2024*, pages 1268–1286. ACM, 2024. doi:10.1145/3670865.3673582.

6 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. doi:10.1016/J.TCS.2020.07.006.

7 Haris Aziz, Ioannis Caragiannis, Ayumi Igarashi, and Toby Walsh. Fair allocation of indivisible goods and chores. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence*, pages 53–59, 2019. doi:10.24963/IJCAI.2019/8.

8 Siddharth Barman, Umang Bhaskar, Yeshwant Pandit, and Soumyajit Pyne. Nearly equitable allocations beyond additivity and monotonicity. In *Thirty-Eighth AAAI Conference on Artificial Intelligence, AAAI 2024, February 20-27, 2024, Vancouver, Canada*, pages 9494–9501. AAAI Press, 2024. doi:10.1609/AAAI.V38I9.28804.

9 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. *Theor. Comput. Sci.*, 1002:114596, 2024. doi:10.1016/J.TCS.2024.114596.

10 Ben Berger, Avi Cohen, Michal Feldman, and Amos Fiat. Almost full EFX exists for four agents. In *Thirty-Sixth AAAI Conference on Artificial Intelligence, AAAI 2022, February 22 - March 1, 2022*, pages 4826–4833. AAAI Press, 2022. doi:10.1609/AAAI.V36I5.20410.

11 Umang Bhaskar, Gunjan Kumar, Yeshwant Pandit, and Rakshitha. Towards envy-freeness relaxations for general nonmonotone valuations. In *Proceedings of the 24th International Conference on Autonomous Agents and Multiagent Systems*, pages 298–306, 2025. doi:10.5555/3709347.3743543.

12 Umang Bhaskar, A. R. Sricharan, and Rohit Vaish. On approximate envy-freeness for indivisible chores and mixed resources. In Mary Wootters and Laura Sanità, editors, *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques, APPROX/RANDOM 2021*, volume 207 of *LIPIcs*, pages 1:1–1:23. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2021. doi:10.4230/LIPICS.APPROX/RANDOM.2021.1.

13 Eric Budish. The Combinatorial Assignment Problem: Approximate Competitive Equilibrium from Equal Incomes. *Journal of Political Economy*, 119(6):1061–1103, 2011.

14 Ioannis Caragiannis, 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, 2019.

15 Hau Chan, Jing Chen, Bo Li, and Xiaowei Wu. Maximin-aware allocations of indivisible goods. In *Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems, AAMAS '19*, pages 1871–1873, 2019. URL: http://dl.acm.org/citation.cfm?id=3331947.

16 Bhaskar Ray Chaudhury, Jugal Garg, and Kurt Mehlhorn. EFX exists for three agents. *J. ACM*, 71(1):4:1–4:27, 2024. doi:10.1145/3616009.

17 Bhaskar Ray Chaudhury, Telikepalli Kavitha, Kurt Mehlhorn, and Alkmini Sgouritsa. A little charity guarantees almost envy-freeness. *SIAM J. Comput.*, 50(4):1336–1358, 2021. doi:10.1137/20M1359134.

18 George Christodoulou, Amos Fiat, Elias Koutsoupias, and Alkmini Sgouritsa. Fair allocation in graphs. In Kevin Leyton-Brown, Jason D. Hartline, and Larry Samuelson, editors, *Proceedings of the 24th ACM Conference on Economics and Computation, EC 2023, London, United Kingdom, July 9-12, 2023*, pages 473–488. ACM, 2023. doi:10.1145/3580507.3597764.

19 Paul W. Goldberg, Kasper Høgh, and Alexandros Hollender. The frontier of intractability for EFX with two agents. In *Algorithmic Game Theory - 16th International Symposium, SAGT 2023, Egham, UK, September 4-7, 2023, Proceedings*, volume 14238 of *Lecture Notes in Computer Science*, pages 290–307. Springer, 2023. doi:10.1007/978-3-031-43254-5_17.

20 Hadi Hosseini, Sujoy Sikdar, Rohit Vaish, and Lirong Xia. Fair and efficient allocations under lexicographic preferences. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 5472–5480, 2021. doi:10.1609/AAAI.V35I6.16689.

21 Vishwa Prakash HV, Pratik Ghosal, Prajakta Nimbhorkar, and Nithin Varma. EFX exists for three types of agents. *CoRR*, abs/2410.13580, 2024. doi:10.48550/arXiv.2410.13580.

22 Alireza Kaviani, Masoud Seddighin, and AmirMohammad Shahrezaei. Almost envy-free allocation of indivisible goods: A tale of two valuations. In *Web and Internet Economics - 20th International Conference, WINE 2024, Proceedings*. Springer, 2024.

23 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*, pages 125–131, 2004. doi:10.1145/988772.988792.

24 Ryoga Mahara. Existence of EFX for two additive valuations. *Discret. Appl. Math.*, 340:115–122, 2023. doi:10.1016/J.DAM.2023.06.035.

25 Benjamin Plaut and Tim Roughgarden. Almost envy-freeness with general valuations. *SIAM J. Discret. Math.*, 34(2):1039–1068, 2020. doi:10.1137/19M124397X.

26 Alkmini Sgouritsa and Minas Marios Sotiriou. On the existence of EFX allocations in multigraphs. In *Proceedings of the 24th International Conference on Autonomous Agents and Multiagent Systems*, pages 2735–2737, 2025. doi:10.5555/3709347.3743995.

27 Alkmini Sgouritsa and Minas Marios Sotiriou. On the existence of EFX allocations in multigraphs. *CoRR*, abs/2502.09777, 2025. doi:10.48550/arXiv.2502.09777.

28 Jinghan A Zeng and Ruta Mehta. On the structure of EFX orientations on graphs. In *Proceedings of the 24th International Conference on Autonomous Agents and Multiagent Systems*, pages 2309–2316, 2025. doi:10.5555/3709347.3743871.

29 Yu Zhou, Tianze Wei, Minming Li, and Bo Li. A complete landscape of EFX allocations on graphs: goods, chores and mixed manna. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence*, pages 3049–3056, 2024. URL: https://www.ijcai.org/proceedings/2024/338.

30 David Zuckerman. Linear degree extractors and the inapproximability of max clique and chromatic number. In *Proceedings of the thirty-eighth annual ACM symposium on Theory of computing*, pages 681–690, 2006. doi:10.1145/1132516.1132612.