EFX Allocations Exist on Triangle-Free Multi-Graphs

Mahyar Afshinmehr
Department of Computer Science,
University of Oxford
Oxford, United Kingdom
mahyar.afshinmehr@cs.ox.ac.uk

Arash Ashuri
Independent Researcher
Tehran, Iran
arashashoori199821@gmail.com

Pouria Mahmoudkhan
Department of Computer Engineering,
Sharif University of Technology
Tehran, Iran
pouriamksh@gmail.com

Kurt Mehlhorn
Max Planck Institute for Informatics,
Saarland Informatics Campus
Saarbrücken, Germany
mehlhorn@mpi-inf.mpg.de

ABSTRACT

We study the fair allocation of indivisible goods among agents, with a focus on limiting envy. A central open question in this area is the existence of *EFX allocations*—allocations in which any envy of any agent i towards any agent j vanishes upon the removal of any single good from j's bundle. Establishing the existence of such allocations has proven notoriously difficult in general, but progress has been made for restricted valuation classes. Christodoulou et al. [31] proved existence for *graphical valuations*, where goods correspond to edges in a graph, agents to nodes, and each agent values only incident edges. The graph was required to be simple, i.e., for any pair of agents, there could be at most one good that both agents value. The problem remained open, however, for *multi-graph valuations*, where for a pair of agents several goods may have value to both. In this setting, Sgouritsa and Sotiriou [52] established existence whenever the shortest cycle with non-parallel edges has length at least six, while Afshinmehr et al. [3] proved existence when the graph contains no odd cycles.

In this paper, we strengthen these results by proving that EFX allocations always exist in multi-graphs that contain no cycle of length three. Assuming monotone valuations, we further provide a pseudo-polynomial time algorithm for computing such an allocation, which runs in polynomial time when agents have cancelable valuations, a strict superclass of additive valuation functions. Accordingly, our results stand as one of the only cases where EFX allocations exist for an arbitrary number of agents.

KEYWORDS

Algorithmic Game Theory; Fair Allocation; Graphs

ACM Reference Format:
Mahyar Afshinmehr, Arash Ashuri, Pouria Mahmoudkhan, and Kurt Mehlhorn. 2026. EFX Allocations Exist on Triangle-Free Multi-Graphs. In *Proc. of the 25th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2026), Paphos, Cyprus, May 25 – 29, 2026*, IFAAMAS, 10 pages. https://doi.org/10.65109/XYOA9688

Proc. of the 25th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2026), C. Amato, L. Dennis, V. Mascardi, J. Thangarajah (eds.), May 25 – 29, 2026, Paphos, Cyprus. © 2026 International Foundation for Autonomous Agents and Multiagent Systems (www.ifaamas.org). https://doi.org/10.65109/XYOA9688

1 INTRODUCTION

Research on *Fair Division* studies how to distribute resources among multiple agents in a way that is considered *fair*. This problem arises in diverse domains, including business asset division, computational resource allocation, course assignments, and dispute resolution, such as divorce settlements and air traffic management ([24, 34, 48, 50, 54]). Since the seminal work of Steinhaus, the field has evolved at the intersection of economics, mathematics, computer science, and social choice theory, becoming a mature area driven by both theoretical and practical relevance. For surveys, see [9, 20, 21, 51].

Over the past decade, research on fair division has primarily focused on allocating indivisible goods, with the central goal of minimizing envy among agents. Since some degree of envy is inevitable in certain instances[1], much of the literature has concentrated on relaxations of exact envy-freeness and on understanding the extent to which these relaxations can be guaranteed or approximated.

The first relaxation of envy-freeness is *envy freeness up to some good* (EF1), introduced formally in [23] and shown achievable earlier in [44]. An allocation is EF1 if for any two agents i, j with assigned bundles X, Y, respectively, there exists some good $g \in Y$ such that i weakly prefers X to $Y \setminus \{g\}$.

A particularly compelling notion is *envy-freeness up to any good (EFX)*, introduced by Caragiannis et al. [27]. An allocation is EFX if removing any good from the bundle allocated to an agent would ensure that no agent envies the remaining bundle. The EFX criterion strikes a delicate balance between fairness and feasibility, and has therefore emerged as a central benchmark in the study of indivisible goods allocation. Despite its appeal, the existence of EFX allocations in general remains a major open problem. Over the last few years, significant progress has been made by identifying valuation classes where EFX allocations are guaranteed to exist.

Today, we know that EFX exists in restricted settings, such as when agents' valuations are identical [49], monotone but only up to two types [45], additive but only up to three types [39], or binary [22]. For three-agent instances, EFX allocations were shown to exist for additive valuations [28], subsequently extended to nice cancelable valuations [17] and MMS-feasible valuations [4].

Christodoulou et al. [31] proved that EFX allocations exist when valuations are represented by a simple graph: goods are edges,

[1] For example, if a single good is more valuable than all others combined, any agent who receives it will inevitably be envied.

agents are nodes, and each agent values only incident edges. This model naturally captures settings such as allocating natural resources among neighboring countries, office spaces among research groups, or public areas among communities.

The natural next step is to consider *multi-graph valuations*, an open question raised by Christodoulou et al. [31], where multiple goods may connect the same pair of agents. This setting is substantially richer and more complex. Understanding whether EFX allocations always exist in this model has emerged as a key open question. Independently, Afshinmehr et al. [3], Bhaskar and Pandit [18], Sgouritsa and Sotiriou [52] established the existence of EFX allocations when the underlying multi-graph contains no odd cycles. Moreover, Sgouritsa and Sotiriou [52] proved that EFX allocations exist whenever the shortest cycle with non-parallel edges has length at least six, while Bhaskar and Pandit [18] further showed that if the underlying graph is t-colorable and its shortest cycle has length at least $2t - 1$, then an EFX allocation is guaranteed. Afshinmehr et al. [3] also proved the existence of EFX allocations on multi-graphs when the underlying graph is a single cycle.

1.1 Our contributions and Techniques

In this work, we answer the open question raised by Christodoulou et al. [31] and strengthen the previously known results in the multi-graph setting, and take a significant step toward resolving the general existence question for multi-graph valuations. Specifically, we prove that EFX allocations always exist when the underlying multi-graph contains no cycle of length three. This substantially improves upon the previous six-cycle bound proved by Sgouritsa and Sotiriou [52], and pushes further the no odd-cycle result proved by Afshinmehr et al. [3]. We present a pseudo-polynomial time algorithm for computing an EFX allocation under monotone valuations. Furthermore, when agents have cancelable valuations, a strict superclass of additive valuations, our algorithm runs in polynomial time (see Theorem 3.1).

In general, our results go strictly beyond the work of Afshinmehr et al. [3], Bhaskar and Pandit [18], Sgouritsa and Sotiriou [52] and expand the frontier of known valuation classes for which EFX allocations are guaranteed to exist, and provide efficient methods for computing them.

Complete Version. Due to space limit of the paper, we defer some of the proofs to the full version [2]. More specifically, we defer the proof of Lemmas 3.6, 3.9, 3.11, and 3.12 to the full version of the paper.

Technical Overview. We give a brief and simplified description of our techniques to prove the existence of EFX allocations on triangle-free multi-graphs. Our Algorithm consists of three phases that move in the space of partial EFX allocations.

Phase One. For each pair of agents i and j, we partition the set of edges between them into two bundles, which we refer to as unit bundles. More precisely, we designate an agent to divide the set of items shared by i and j into two EFX-feasible[2] bundles from her perspective. These unit bundles remain fixed throughout the algorithm: if an agent receives one item from a unit bundle, she receives the entire bundle.

In Phase One, each agent i is assigned exactly one of her incident unit bundles, subject to the following conditions:

(1) The resulting partial allocation is EFX.
(2) Each agent weakly prefers her own bundle to every unallocated incident unit bundle.
(3) The longest path in the envy graph[3] has length at most one.

To ensure these properties, agents select their most preferred unallocated incident unit bundle in a carefully chosen order. Both the selection order and the construction of unit bundles are essential for maintaining the above invariants. For instance, if agent i comes before agent j in this order, then agent j cuts the set $E(i, j)$[4] into two EFX-feasible bundles for herself, that are the unit bundles between agents i and j.

If, at the end of Phase One, no agent envies another, then the remaining unit bundles can be allocated as follows: for each pair of agents i and j, one of the unit bundles between them is assigned to i, and the other to j, while ensuring that each agent retains the unit bundle assigned to her in Phase One. This yields a complete EFX allocation. The justification is that, for every pair (i, j), the value of j's bundle from i's perspective equals the value of the unit bundle incident to both i and j that is allocated to j. By construction, this value does not exceed the value of i's own bundle—since that unit bundle is either (a) unallocated after Phase One, in which case i does not envy it, or (b) allocated to j, in which case i does not envy it by assumption.

If, however, some agents are still envied after Phase One, then every non-envied agent i can nevertheless be assigned one of her incident unit bundles in $E(i, j)$ for every other agent j while preserving the EFX property, by the same argument. The remaining challenge lies in allocating the unassigned unit bundles incident to envied agents.

Phases Two and Three. The second and third phases of the algorithm are designed to allocate these remaining unit bundles that are incident to envied agents. After phase one, the envy graph would look like a set of stars since the length of its longest path is at most one.

Consider an arbitrary star centered at a non-envied agent i_0, who envies agents i_1, i_2, and also consider some additional agent j (see Fig. 1). Let (C_1, C_2) denote the unit bundles between agents i_1 and j. Similarly, let (D_1, D_2) denote the unit bundles between agents i_2 and j. We attempt to allocate the remaining incident unit bundles C_1 and D_1 to agent i_0, allowing the allocation to assign an item to a non-incident agent.

Note that agent j does not envy i_0's original bundle X_{i_0}, since the longest path in the envy graph has length one. Therefore, if j does not envy the combined bundle $C_1 \cup D_1$, then she also does not envy $C_1 \cup D_1 \cup X_{i_0}$. This follows from the triangle-free property of the multi-graph, which ensures that at least one of the sets $C_1 \cup D_1$ or $X_{i_0} \cap E(i_0, j)$ is empty.

If the attempt of allocating C_1 and D_1 to agent i_0 fails to maintain an EFX partial allocation, we adjust the allocation greedily. For

[2]Given a partition $\mathbf{Y} = (Y_1, \cdots, Y_l)$ of a set S, we say a bundle Y_k is EFX-feasible for an agent i, if i weakly prefers Y_k to every other bundle in the partition after the removal of any item from it.

[3]The envy graph associated with a partial allocation has one node per agent, and an edge from agent i to agent j whenever i envies j's bundle relative to her own.

[4]$E(i, j)$ denotes the set of items (edges) between agents i and j.

instance, if $C_1 \cup D_1$ is more valuable than agent j's current bundle, we instead allocate $C_1 \cup D_1$ to j. Furthermore, after adding these unit bundles for every possible j, if agent i_1 comes to envy i_0, we swap the bundles of i_0 and i_1, thereby making both agents happier.

The second and third phases of our algorithm consist of a sequence of such local adjustments—allocations and unallocations of unit bundles—designed to preserve the EFX property while ensuring that all items are eventually allocated.

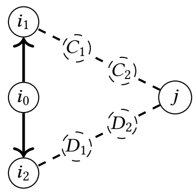

Figure 1: This figure shows a star in the envy graph, and an arbitrary agent j, and it depicts unit bundles among envied agents of this star and agent j.

1.2 Further Related Works

In another line of research, Amanatidis et al. [10] showed that every additive multi-graph instance admits a $\frac{2}{3}$-EFX allocation, and Kaviani et al. [42] recently improved this bound to $\frac{1}{\sqrt{2}}$-EFX. Furthermore, Kaviani et al. [42] showed that EFX allocations exists on multi-graphs if the valuation functions are restricted additive. Zhou et al. [57] studied the mixed manna setting with both goods and chores, proving that deciding the existence of EFX orientations on simple graphs with additive valuations is NP-complete. Zeng and Mehta [55] established a connection between the existence of EFX orientations and the chromatic number of the graph. Blazej et al. [19] refined this complexity landscape by showing that bipartiteness is a tight boundary for tractability, identifying hardness even for graphs very close to bipartite. More recently, Deligkas et al. [33] proved that EF1 orientations always exist for monotone valuations.

For more general settings, a lot of work has focused on relaxations of EFX. One such relaxation has aimed to achieve multiplicative approximations of EFX. Plaut and Roughgarden [49] showed the existence of $1/2$-EFX allocations for subadditive valuations, and Amanatidis et al. [11] proved the existence of $1/\phi \approx 0.618$-EFX allocations for additive valuations. $2/3$-EFX allocations were shown to exist for more restrictive settings like when there are at most four types of valuations [40], at most seven agents [10], and when the agents agree on what are the top n items [47]. Barman et al. [15] achieved improved EFX approximations for restricted settings.

A related relaxation, EF2X, requires that envy disappear after the removal of any two goods from any bundle. EF2X allocations were shown to exist for four agents with cancelable valuations [13], any number of agents with restricted additive valuations [8], and with $(\infty, 1)$-bounded valuations [43]. Several other relaxations of EFX have also been studied, such as EFL [14], EFR [35], EEFX and MXS [7, 25]. In addition, several works have explored the simultaneous satisfaction of multiple fairness notions. For instance, Akrami and Rathi [6] established the coexistence of $2/3$-MMS and EF1, Ashuri and Gkatzelis [12] combined MXS with EFL, and Feige [36] achieved RMMS alongside EFL. Recently, Akrami et al. [5] showed the coexistence of EEFX and EFL.

Another line of research has focused on "partial allocations," donating some of the goods to charity and achieving EFX with the rest. This was first studied by Caragiannis et al. [26] who showed the existence of a partial allocation that satisfies EFX and its Nash social welfare is half of the maximum possible. EFX allocations were shown to exist up to $n-1$ donated goods [30], subsequently improved to $n-2$ goods [17, 45]. Berger et al. [17] further showed that an EFX allocation exists for 4 agents with at most one donated good. The number of donated goods was improved at the expense of achieving $(1-\varepsilon)$-EFX, instead of exact EFX [4, 16, 29, 41].

Another line of research is on indivisible chores, where each agent is associated with a cost function. The following approximations of EFX were shown to exist: $O(n^2)$-EFX [56], 4-EFX [38], 2-EFX for three agents [1, 32], and recently 2-EFX for any number of agents [37]. Surprisingly, Christoforidis and Santorinaios [32] demonstrated that for monotone cost functions, there exist instances with no EFX allocation. Another major open question—whether allocations can be both EF1 and PO—was recently resolved affirmatively by Mahara [46].

2 PRELIMINARIES

An instance of discrete fair division is a tuple $\langle N, M, \{v_i\}_{i\in[n]}\rangle$, where $N = [n] = \{1, 2, \dots, n\}$ is a set of agents, M is a set of m indivisible goods, and $\{v_i\}_{i\in[n]}$ is a profile of valuation functions, where $v_i : 2^M \to \mathbb{R}_{\geq 0}$ for each agent $i \in N$ determines i's value for each subset of goods. For notational simplicity, for a subset of goods S and good $g \in S$, we use $S \setminus g$ to denote $S \setminus \{g\}$.

We consider valuation functions that are monotone, i.e., for any $S \subseteq T$, $v(S) \leq v(T)$. The valuation function v is additive if, for any subset of items S, we have $v(S) = \Sigma_{g\in S} v(\{g\})$. The valuation function v is cancelable if, for any $S, T \subseteq M$ and any $g \in M \setminus (S \cup T)$, if $v(S \cup \{g\}) > v(T \cup \{g\})$, then $v(S) > v(T)$, i.e., removing the same good from two bundles would not change the relative preference between the two. Note that the class of cancelable valuation functions is a strict superclass of additive valuations.

Multi-graph Instances. A fair division instance on a *multi-graph*[5], $\mathcal{I} = \langle [n], [m], \{v_i\}_{i\in[n]}\rangle$, is represented by a multi-graph $G = (V, E)$, where the n agents correspond to the vertices in V, and the m goods correspond to the edges in E. The structure is such that for every agent $i \in [n]$ and every subset of goods $S \subseteq [m]$,$v_i(S) = v_i(S \cap E(i))$, where $E(i) \subseteq E$ is the set of goods incident to i, and $E(i, j)$ the set of edges between i and j. For a multi-graph $G = (V, E)$, we define its skeleton as a graph $G' = (V, E')$ where G' has the same set of vertices, and there is a single edge between two vertices if they share at least one edge in G, i.e., i is connected to j in G' if $E(i, j) \neq \emptyset$ in G. We call a multi-graph *triangle-free* if its skeleton does not contain any cycle of length three, i.e., the girth[6] of its skeleton is greater than or equal to four. In this work, we use the words *good*, *item*, and *edge* interchangeably.

Allocations and Orientations. A partial allocation is an ordered tuple of disjoint subsets of $[m]$ denoted by $\mathbf{X} = \langle X_1, \dots, X_n\rangle$, i.e, for every pair of distinct agents i and j, we have $X_i, X_j \subseteq [m]$ and $X_i \cap X_j = \emptyset$. Here, X_i denotes the bundle allocated to agent $i \in [n]$

[5] A multi-graph may contain multiple edges between two vertices.
[6] The girth of a graph is the length of its shortest cycle.

in $\mathbf{X}$. A partial orientation is a partial allocation where $\forall i \in [n]$, we have $X_i \subseteq E(i)$. We say an allocation $\mathbf{X}$ is complete if $\bigcup_{i \in [n]} X_i = [m]$.

Envy, Strong Envy, EFX-Feasibility, EFX Allocation, EFX Cut, and EFX Orientation. We say an agent i envies a bundle T with respect to bundle S if $v_i(T) > v_i(S)$. Moreover, agent i *strongly envies* T with respect to S if there exists a good $g \in T$ such that $v_i(S) < v_i(T \setminus g)$. Given an allocation $\mathbf{X}$, we say agent i envies (respectively, strongly envies) agent j if i envies (strongly envies) bundle X_j with respect to bundle X_i. We say an allocation is EFX if there is no strong envy between any pair of agents. Given a partition $\mathbf{Y} = (Y_1, \dots, Y_k)$ of a subset of goods into k bundles, we say bundle Y_ℓ is EFX-feasible for agent i in $\mathbf{Y}$ if agent i does not strongly envy any of the bundles in $\mathbf{Y}$ with respect to Y_ℓ. We say the partition $\mathbf{C} = (C_1, C_2)$ of items in set S is an EFX cut of set S for agent i if both C_1 and C_2 are EFX-feasible for agent i in $\mathbf{C}$. For a fair division instance on a multi-graph, the allocation $\mathbf{X}$ is an EFX Orientation if it is an EFX allocation and $X_i \subseteq E(i)$ for every $i \in [n]$.

Next, we demonstrate a property of EFX orientations on multi-graphs via the following observation.

Observation 2.1. *For a multi-graph instance, consider a partial EFX orientation $\mathbf{X}$ where an agent i is envied by one of her neighbors j. Then, we must have $X_i \subseteq E(i, j)$. In particular, any agent is envied by at most one agent in any EFX orientation.*

3 EFX ALLOCATIONS ON TRIANGLE-FREE MULTI-GRAPHS

In this section, we prove the main result of our work (Theorem 3.1), the existence of an EFX allocation for triangle-free multi-graphs for an arbitrary number of goods and agents with monotone valuations.

Theorem 3.1. *For every multi-graph instance, EFX allocations always exist when the skeleton of the underlying multi-graph contains no cycle of length three, and we can compute one in pseudo-polynomial time when the valuation functions are monotone and in polynomial time when the valuation functions are cancelable.*

Our proof consists of three main phases: in the first two phases, we deal with partial orientations. We will define a set of properties in each phase and try to satisfy all of them. Note that in the second phase, we aim to add properties to the output orientation of the first phase while preserving the properties defined in the first phase. Finally, in the third phase, we allocate the remaining items to a third party, changing our orientation to an allocation, and complete our EFX allocation. It is notable that the first two phases can be applied to any multi-graph instance; i.e., we use the triangle-free condition only in the final phase. We will now introduce each phase and prove our main result in the following subsections.

3.1 Phase One

We generalize the concept of cut configurations, introduced by Afshinmehr et al. [3], to the setting of all multi-graphs. Accordingly, their definition relied strongly on the structure of a bipartite multi-graph for which they showed the existence of an EFX allocation. We define the concept of configuration for every pair of vertices in any multi-graph as follows:

	$A_{i,j}(\mathbf{X}, \sigma)$	$A_{j,i}(\mathbf{X}, \sigma)$
$X_i \cap E(i,j) = \emptyset, X_j \cap E(i,j) = \emptyset$	$\arg\max_{C \in \{C_{j,i}^{(1)}, C_{j,i}^{(2)}\}} \{v_i(C)\}$	$\arg\max_{C \in \{C_{j,i}^{(1)}, C_{j,i}^{(2)}\}} \{v_j(C)\}$
$X_i \cap E(i,j) = \emptyset, X_j \cap E(i,j) \neq \emptyset$	$E(i,j) \setminus X_j$	$\emptyset$
$X_i \cap E(i,j) \neq \emptyset, X_j \cap E(i,j) = \emptyset$	$\emptyset$	$E(i,j) \setminus X_i$
$X_i \cap E(i,j) \neq \emptyset, X_j \cap E(i,j) \neq \emptyset$	$\emptyset$	$\emptyset$

Table 1: Definition of $A_{i,j}(\mathbf{X}, \sigma)$ and $A_{j,i}(\mathbf{X}, \sigma)$ when $i \prec_\sigma j$ and the j-*cut* configuration is fixed between agents i and j.

Definition 3.2 (Cut Configuration)**.** *For every pair of agents i and j, we fix two partitions of the set $E(i,j)$. The partition $(C_{i,j}^{(1)}, C_{i,j}^{(2)})$ of $E(i,j)$ is an EFX cut for agent i and is called the i-*cut *configuration between agents i and j. Similarly, the j-*cut *configuration is the partition $(C_{j,i}^{(1)}, C_{j,i}^{(2)})$ of $E(i,j)$ that is an EFX cut for agent j. Note that the first index indicates which agent makes the cut.*

We aim to fix a specific configuration between every pair of vertices in our proof. We consider a sequence of agents denoted by $\sigma = [\sigma(1), \sigma(2), \cdots, \sigma(n)]$, where $\sigma(i) \in [n]$. Note that the sequence σ is a permutation of agents in $[n]$. Every such sequence σ imposes a fixed set of configurations on our input instance. More precisely, if i precedes j in σ (denoted by $i \prec_\sigma j$), then we use the j-cut configuration for the set $E(i,j)$ in our construction. Formally, we define *unit bundles* as follows:

Definition 3.3 (Unit Bundles)**.** *For every sequence σ of agents, if $i \prec_\sigma j$, we define the bundles $C_{j,i}^{(1)}$ and $C_{j,i}^{(2)}$ of the j-*cut *configuration between agents i and j as the* unit bundles *between these two agents. We say a unit bundle C is incident for agent i, if C is a unit bundle of $E(i,j)$ for some agent j.*

Thus, we always deal with a sequence that has fixed configurations in our graph. In this phase, we aim to find an appropriate sequence that satisfies certain properties. We use the concept of *available* edges for an agent, introduced by Afshinmehr et al. [3], during our proof several times.

Definition 3.4 ($A_{i,j}(\mathbf{X}, \sigma)$ and $A_{j,i}(\mathbf{X}, \sigma)$)**.** *For a partial orientation $\mathbf{X} = \langle X_1, \cdots, X_n \rangle$ and a sequence σ, and any two agents i and j where $i \prec_\sigma j$, we define the sets $A_{i,j}(\mathbf{X}, \sigma) \subseteq E(i,j)$ and $A_{j,i}(\mathbf{X}, \sigma) \subseteq E(i,j)$ in four cases using Table 1. We call $A_{i,j}(\mathbf{X}, \sigma)$ the set of* available *edges in $E(i,j)$ for agent i, and similarly, $A_{j,i}(\mathbf{X}, \sigma)$ the set of* available *edges in $E(i,j)$ for agent j.*

In our construction, for every pair of agents i and j, we allocate at most one of the unit bundles in $E(i,j)$ to agent i. Accordingly, $A_{i,j}(\mathbf{X}, \sigma)$ is in fact the most valuable unallocated unit bundle that agent i can get from the set $E(i,j)$ subject to the mentioned restriction. Moreover, if an agent like i receives one item from a unit bundle, she receives the entire bundle, i.e., for an agent i and unit bundle C, if there exists a good $g \in X_i \cap C$, then $C \subseteq X_i$.

Next, we introduce the key properties that we aim to satisfy in the first phase of our algorithm.

Key Properties: In this phase, we search for a partial allocation $\mathbf{X} = \langle X_1, \cdots, X_n \rangle$ and a sequence $\sigma = [\sigma(1), \cdots, \sigma(n)]$ satisfying the following properties:

(1) $\mathbf{X}$ is an EFX orientation.
(2) For any two agents i and j where $i \prec_\sigma j$, the items in $E(i,j)$ must be allocated according to the j-*cut* configuration

$(C^{(1)}_{j,i}, C^{(2)}_{j,i})$ to either one of their endpoints. Formally, one of the following must hold in $\mathbf{X}$: either no item in $E(i, j)$ is allocated or one of the unit bundles in the j-cut configuration is allocated to either i and j and the other unit bundle is unallocated or both unit bundles in the j-cut configuration are allocated, one to i and one to j. Note that additional items may be allocated to i or j.

(3) For any agent i and any unallocated unit bundle C, we have $v_i(X_i) \geq v_i(C)$.

(4) The length of any envy path[7] in the graph is at most one.

We now prove that there exists an allocation $\mathbf{X}$ and sequence σ satisfying all four properties simultaneously.

3.1.1 Satisfying Properties (1)-(4). We design an iterative algorithm to compute a partial allocation $\mathbf{X}$ and sequence σ simultaneously such that they satisfy properties 1-4.

In each step of our algorithm, we use two disjoint partial sequences σ_L and σ_R together with a set U containing agents that are not in any of these sequences. Furthermore, $\mathbf{X} = \langle X_1, \cdots, X_n \rangle$ denotes the partial allocation in each step. More precisely, we will gradually build the sequence σ from both sides by appending agents to the end of the sequence σ_L or to the beginning of the sequence σ_R. This means that in each step, our final sequence σ has the form of $\sigma = [\sigma_L(1), \cdots, \sigma_L(|\sigma_L|), u_1, \cdots, u_{|U|}, \sigma_R(1), \cdots, \sigma_R(|\sigma_R|)]$ where $u_i \in U$. Basically, the exact place of agents outside of U is determined, and we only do not know the ordering of agents in U in our final sequence. Note that the configuration between any agent $i \in [n] \setminus U$ and any other agent j is fixed in any step of the algorithm. Also, for an agent $i \in U$, the bundle X_i is not yet determined. In each step, we will gradually build σ by fixing the position of some agents in U in our sequence and allocating a unit bundle to each of them, and then remove them from U.

We begin with $\sigma_L = \sigma_R = []$ and $U = [n]$. In each step, we will decrease $|U|$ by at least one, while maintaining the following invariants:

(i) Every agent in σ_L and σ_R has received a unit bundle that is incident to her and another agent in σ_L or σ_R, that is, no agent in these two partial sequences has received any item incident to agents in U.
(ii) For any agent $i \in U$, we have $X_i = \emptyset$.
(iii) The allocation $\mathbf{X}$ is an orientation.
(iv) Agents in σ_L and σ_R do not strongly envy any bundle in $\mathbf{X}$.
(v) Agents in σ_L do not envy any agent in $\mathbf{X}$.
(vi) Agents in σ_R do not envy each other.
(vii) For any agent $i \notin U$ and any other agent $j \in [n]$ property 2 holds. Additionally, for any agent $i \notin U$, property 3 holds.

In the beginning, we have $\sigma_L = \sigma_R = []$ and $U = [n]$, which clearly satisfy all invariants. Thus, it remains to show that we can move in the space of partial allocations satisfying invariants (i)-(vii) while decreasing the number of agents in U.

Now, we go over a single step of our algorithm and show that we can decrease $|U|$ while maintaining all invariants. Consider a single step of our algorithm with $U \neq \emptyset$. Note that the configuration between any agent in U and any agent in $[n] \setminus U$ is fixed before this step. Let i be an arbitrary agent in U. We will append i to the beginning of σ_R, and remove i from U. Therefore, for any agent $u \in U$, the configuration between u and i will be fixed to *i-cut*. Thereby, we have all the configurations in which agent i is involved in. Although σ is not complete yet, with a slight abuse of notation, we use the definition of $A_{i,j}(\mathbf{X}, \sigma)$. This is indeed valid, because the unit bundles incident to agent i are fixed. Let $j = \arg\max_{j \in [n]} v_i(A_{i,j}(\mathbf{X}, \sigma))$. If $j \notin U$, we allocate $A_{i,j}(\mathbf{X}, \sigma)$ to agent i and stop. Otherwise, we add j to the end of σ_L and remove j from U, thereby fixing configurations involving j. Let $k = \arg\max_{k \in [n]} v_j(A_{j,k}(\mathbf{X}, \sigma))$. There are three cases:

- **Case 1:** $k = i$. We allocate $A_{j,i}(\mathbf{X}, \sigma)$ to agent j and then repeat the procedure for agent i, that is we will again find an agent $j = \arg\max_{j \in [n]} v_i(A_{i,j}(\mathbf{X}, \sigma))$ and continue.
- **Case 2:** $k \neq i$ **and** $k \notin U$. We allocate $A_{j,k}(\mathbf{X}, \sigma)$ to agent j and $A_{i,j}(\mathbf{X}, \sigma)$ to agent i and stop.
- **Case 3:** $k \neq i$ **and** $k \in U$. We allocate $A_{j,k}(\mathbf{X}, \sigma)$ to agent j and $A_{i,j}(\mathbf{X}, \sigma)$ to agent i. Then, we add k to the beginning of σ_R and then repeat the procedure that we did for agent i exactly for agent k, that is we will again find an agent $j = \arg\max_{j \in [n]} v_k(A_{k,j}(\mathbf{X}, \sigma))$ and continue.

Formally, the procedure is defined using Algorithm 1. Lemma 3.5 shows that this procedure maintains our seven invariants and decreases $|U|$.

Lemma 3.5. *Algorithm 1 holds invariants (i)-(vii) and decreases $|U|$.*

Proof. First, note that the procedure defined by Algorithm 1 terminates since it decreases $|U|$ by at least one. By construction, whenever an agent receives a unit bundle, it has already been removed from U and been added to either σ_L or σ_R. Moreover, whenever an agent receives a unit bundle incident to an agent in U, this agent will be removed from U and will be added to either σ_L or σ_R. Also, agents only receive incident unit bundles. Therefore, invariants (i), (ii), and (iii) hold. Every agent $i \notin U$ receives only a single unit bundle. Accordingly, whenever such an agent receives a unit bundle, it is chosen greedily among all of her unallocated incident unit bundles, meaning that invariant (vii) is maintained. Therefore, we need to prove that invariants (iv)-(vi) hold.

Note that during Algorithm 1, agents that are added to σ_R are denoted by i, and agents that are added to σ_L are denoted by j.

First, we show that Algorithm 1 maintains invariant (v). Whenever an agent j is added to the end of σ_L, she greedily picks a unit bundle. Note that whenever agent j receives a unit bundle, the other agents in σ_L or σ_R have not yet been assigned a unit bundle incident to agent j by invariant (i), so agent j does not envy any other agent. By invariant (vii), agents previously in σ_L do not envy the unit bundle that j is receiving. Therefore, invariant (v) holds.

Next, we show that Algorithm 1 maintains invariant (vi). Whenever an agent i is added to σ_R and eventually receives a unit bundle, she does not envy agents who were previously added to σ_R since they are not receiving a unit bundle incident to agent i by invariant (i). Moreover, agents previously in σ_R do not envy agent i by invariant (vii). Therefore, invariant (vi) also holds.

Finally, we show that Algorithm 1 maintains invariant (iv). Note that by invariant (v), we only need to show that agents in σ_R do not strongly envy any bundle in $\mathbf{X}$. Consider an agent j in σ_L

[7] $i_0 \to i_1 \to \cdots \to i_l$ is an envy path if each agent i_q envies the agent i_{q+1}.

and an agent i in σ_R. If $X_j \cap E(i,j) = \emptyset$, then, agent i does not envy agent j. Otherwise, agent j has received a unit bundle from $E(i,j)$ and the configuration between agents i and j is fixed to *i-cut*. Therefore, whenever agent i receives a unit bundle, $A_{i,j}(\mathbf{X},\sigma)$ is the unit bundle in $E(i,j)$ that is not allocated to j, and since agent i picks an available unit bundle incident to her greedily, she receives a unit bundle with a value at least as much as $A_{i,j}(\mathbf{X},\sigma)$ to her. Thus, by the definition of our configurations, agent i will not strongly envy agent j, meaning that invariant (iv) also holds. □

Algorithm 1: Sequence Augmentation

Input: Partial allocation $\mathbf{X}$, sequences σ_L and σ_R, and set U satisfying invariants (i)-(vii).

Output: Partial allocation $\mathbf{X}$, sequences σ_L and σ_R, and set U satisfying invariants (i)-(vii) with strictly decreased $|U|$.

$i \leftarrow$ an arbitrary vertex in U
$\sigma_R \leftarrow [i, \sigma_R(1), \cdots, \sigma_R(|\sigma_R|)]$
$U \leftarrow U \setminus \{i\}$
$j \leftarrow \arg\max_{j \in [n]} v_i(A_{i,j}(\mathbf{X},\sigma))$
while $j \in U$ **do**
  $\sigma_L \leftarrow [\sigma_L(1), \cdots, \sigma_L(|\sigma_L|), j]$
  $U \leftarrow U \setminus \{j\}$
  $k \leftarrow \arg\max_{k \in [n]} v_j(A_{j,k}(\mathbf{X},\sigma))$
  if $k = i$ **then**
    $X_j \leftarrow A_{j,i}(\mathbf{X},\sigma)$
    $j \leftarrow \arg\max_{j \in [n]} v_i(A_{i,j}(\mathbf{X},\sigma))$
  else if $k \notin U$ **then**
    $X_j \leftarrow A_{j,k}(\mathbf{X},\sigma)$
    break
  else if $k \in U$ **then**
    $X_j \leftarrow A_{j,k}(\mathbf{X},\sigma)$
    $X_i \leftarrow A_{i,j}(\mathbf{X},\sigma)$
    $i \leftarrow k$
    $\sigma_R \leftarrow [i, \sigma_R(1), \cdots, \sigma_R(|\sigma_R|)]$
    $U \leftarrow U \setminus \{i\}$
    $j \leftarrow \arg\max_{j \in [n]} v_i(A_{i,j}(\mathbf{X},\sigma))$
$X_i \leftarrow A_{i,j}(\mathbf{X},\sigma)$
return $\mathbf{X}, \sigma_L, \sigma_R, U$

Now, we are able to prove the main Lemma of this section:

Lemma 3.6. *There exists a complete sequence σ and a partial allocation $\mathbf{X}$ satisfying properties 1-4.*

Now that we have proved the existence of an allocation $\mathbf{X}$ and sequence σ satisfying properties 1-4, we have completed our first phase. Before moving to the second phase, we give an intuitive observation on the sequence σ:

Observation 3.7. *Let σ be the output sequence for phase one. We construct an allocation $\mathbf{X}'$ as follows: we fix the configurations imposed by σ, i.e., for any pair of agents i and j where $i \prec_\sigma j$, we use the j-cut configuration between agents i and j. Then, we run a greedy algorithm using σ as a picking sequence. More precisely, the greedy algorithm consists of n iterations, where in each iteration k, agent $\sigma(k)$ will greedily pick the most preferred unallocated unit bundle for her. The output of such a greedy algorithm using σ as a picking sequence is identical to the allocation $\mathbf{X}$, which was the output of phase one, i.e., $\mathbf{X}' = \mathbf{X}$.*

3.2 Phase Two

Let the allocation $\mathbf{X}$ and sequence σ be the outputs of the previous phase. The sequence σ will be fixed for the rest of our algorithm. We aim to add three more properties to the allocation $\mathbf{X}$. We begin by defining some useful notation and then introduce three new properties. Since σ is fixed, we will drop it from our notation.

3.2.1 *Some Useful Notations.*

For a partial allocation $\mathbf{X}$ and a sequence σ satisfying properties 1-4, we introduce the following definitions:

- Given any pair of agents i, j, we define $B^{(1)}_{i,j}(\mathbf{X})$, $B^{(2)}_{i,j}(\mathbf{X})$, $B^{(1)}_{j,i}(\mathbf{X})$, and $B^{(2)}_{j,i}(\mathbf{X})$ as follows:
  - If both of the unit bundles in $E(i,j)$ are unallocated, then $B^{(1)}_{i,j}(\mathbf{X}) = B^{(2)}_{j,i}(\mathbf{X})$, and they are one of the unit bundles in $E(i,j)$, and $B^{(2)}_{i,j}(\mathbf{X}) = B^{(1)}_{j,i}(\mathbf{X})$ are the other unit bundle in $E(i,j)$.
  - If agent i possess a unit bundle in $E(i,j)$, then $B^{(1)}_{i,j}(\mathbf{X}) = B^{(2)}_{i,j}(\mathbf{X}) = \emptyset$, and the same follows for agent j.
  - If agent i does not possess a unit bundle in $E(i,j)$, and j possess a unit bundle in $E(i,j)$, then $B^{(1)}_{i,j}(\mathbf{X}) = B^{(2)}_{i,j}(\mathbf{X})$ and they are the unallocated unit bundle in $E(i,j)$, and the same follows for agent j.
- For $i \in [n]$, let $B^{(1)}_i(\mathbf{X}) = \bigcup_{j \in [n] \setminus \{i\}} B^{(1)}_{i,j}(\mathbf{X})$ and $B^{(2)}_i(\mathbf{X}) = \bigcup_{j \in [n] \setminus \{i\}} B^{(2)}_{i,j}(\mathbf{X})$.
- For $i \in [n]$, $U_i(\mathbf{X})$ is the set of all unallocated edges incident to i. Note that $B^{(1)}_i(\mathbf{X}), B^{(2)}_i(\mathbf{X}) \subseteq U_i(\mathbf{X})$.

Next, we introduce our new properties that we wish to add to our partial allocation:

Key Properties: We search for a partial allocation $\mathbf{X} = \langle X_1, \cdots, X_n \rangle$ that satisfies the following properties in addition to properties 1-4:

(5) For any non-envied agent $i \in [n]$, we have $B^{(1)}_i(\mathbf{X}) = \emptyset$.
(6) For any non-envied agent $i \in [n]$, we have $v_i(X_i) \geq v_i(U_i(\mathbf{X}))$
(7) For any envied agent $i \in [n]$, if j envies i, we have:

$$v_i(X_i) \geq v_i(X_j \cup B^{(1)}_i(\mathbf{X})) \text{ and } v_i(X_i) \geq v_i(X_j \cup B^{(2)}_i(\mathbf{X})).$$

We will now introduce an algorithm that satisfies these properties by allocating or unallocating some unit bundles to agents. The procedure is formally defined in Algorithm 2. The algorithm, first checks if there exists an agent i for whom property 5 does not hold, i.e., $B^{(1)}_i(\mathbf{X}) \neq \emptyset$, then it allocates $X_i \cup B^{(1)}_i(\mathbf{X})$ to agent i. Whenever that property 5 holds, Algorithm 2 checks if there exists an agent i for whom property 6 does not hold, i.e., $v_i(X_i) < v_i(U_i(\mathbf{X}))$. Then, for every j which there exists an unallocated unit bundle in $E(i,j)$, agent i swaps it with her unit bundle in $E(i,j)$, leaving the unit bundle from every other j untouched. Note that the previous $U_i(\mathbf{X})$ is now a subset of the new X_i. Whenever that properties 5 and 6 hold, Algorithm 2 checks if there exists an agent i for whom property 7 does not hold, so if agent j envies i, then $v_i(X_i) < v_i(X_j \cup B_i^{(u)}(\mathbf{X}))$ for some $u \in \{1, 2\}$. Then, agents i and j swap the unit bundles in $E(i, j)$ with each other, and $X_i \cup B_i^{(u)}(\mathbf{X})$ is allocated to agent i.

After the termination of our algorithm, by its definition, properties 5-7 hold. Thus, we need to show that algorithm terminates in polynomial time while maintaining the first four properties. We will show that each if statement preserves our first four properties.

Algorithm 2: Satisfying (5)-(7)

Input: $(\mathbf{X}, \sigma)$ satisfying properties 1-4
Output: Allocation $\mathbf{X}$ satisfying properties 1-7
while *at least one of properties (5)-(7) is not satisfied* **do**
  if *there exists a non-envied agent* $i \in [n]$ *such that* $B_i^{(1)}(\mathbf{X}) \neq \emptyset$ **then**
    $X_i \leftarrow X_i \cup B_i^{(1)}(\mathbf{X})$
  else if *there exists a non-envied agent* $i \in [n]$ *such that* $v_i(X_i) < v_i(U_i(\mathbf{X}))$ **then**
    let $T = \{j : E(i, j) \cap U_i(\mathbf{X}) = \emptyset\}$
    $X_i \leftarrow (X_i \cap \bigcup_{j \in T} E(i, j)) \cup U_i(\mathbf{X})$
  else if *there exists an agent* i *envied by* j *such that* $v_i(X_i) < v_i(X_j \cup B_i^{(u)}(\mathbf{X}))$ *for some* $u \in \{1, 2\}$ **then**
    Let (C_1, C_2) be the unit bundles of $E(i, j)$.
    Swap unit bundles C_1 and C_2 between agents i and j.
    $X_i \leftarrow X_i \cup B_i^{(u)}(\mathbf{X})$
return X

We now introduce a potential function to show that our algorithm terminates after a polynomial number of iterations. Let $\phi_1(\mathbf{X})$ be the number of envied agents, $\phi_2(\mathbf{X})$ be the number of non-envied agents violating property 6, and $\phi_3(\mathbf{X})$ be the number of non-envied agents breaking property 5. We define the lexicographic potential function $\phi(\mathbf{X})$ to be

$$\phi(\mathbf{X}) = \langle \phi_1(\mathbf{X}), \phi_2(\mathbf{X}), \phi_3(\mathbf{X}) \rangle,$$

which we will show that it decreases in lexicographic order in each iteration. Lemmas 3.9, 3.11, and 3.12.

Observation 3.8. *If the allocation* $\mathbf{X}$ *and sequence* σ *satisfy property 2, then for all agents* i *and* j*, we have that* $B_{i,j}^{(1)}(\mathbf{X}), B_{i,j}^{(2)}(\mathbf{X}) \in \{C^{(1)}, C^{(2)}, \emptyset\}$*, where* $C^{(1)}, C^{(2)}$ *are the unit bundles between* i *and* j*. Moreover, if* $(\mathbf{X}, \sigma)$ *satisfy property 3, too, then for any pair of agents* i *and* j*, we have* $v_i(X_i) \geq v_i(B_{i,j}^{(1)}(\mathbf{X}))$ *and* $v_i(X_i) \geq v_i(B_{i,j}^{(2)}(\mathbf{X}))$.

Lemma 3.9. *During the execution of Algorithm 2, whenever the first if-statement is executed, the potential function* $\phi(\mathbf{X})$ *strictly decreases while properties 1-4 remain satisfied.*

Observation 3.10. *If allocation* $\mathbf{X}$ *and sequence* σ *satisfy properties 1-5, then looking at the allocated edges we find*

- *Between two non-envied agents* i *and* j*, both unit bundles must be allocated by property 5.*
- *Between envied agent* i *and agent* j *who is the sole agent who envies* i*, both bundles must also be allocated. This is because* i *must possess some unit bundle that* j *envies, and due to property 4,* j *must be non-envied, so she has the remaining unit bundle by property 5.*
- *Between envied agent* i *and non-envied agent* j *who does not envy* i*, there must be a single unallocated unit bundle. Agent* i *cannot possess any unit bundle from* $E(i, j)$*; otherwise, she would be strongly envied. Also,* j *must have a single unit bundle from* $E(i, j)$*, by property 5.*
- *Between two envied agents* i *and* j*, both unit bundles must be unallocated.*

Lemma 3.11. *During the execution of Algorithm 2, whenever the second if-statement is executed, the potential function* $\phi(\mathbf{X})$ *strictly decreases while properties 1-4 remain satisfied.*

Lemma 3.12. *During the execution of Algorithm 2, whenever the third if-statement is executed, the potential function* $\phi(\mathbf{X})$ *strictly decreases while properties 1-4 remain satisfied.*

Given that $\phi_1(\mathbf{X})$, $\phi_2(\mathbf{X})$ and $\phi_3(\mathbf{X})$ are each at most n, Algorithm 2 satisfies properties 1-7 in at most n^3 iterations.

3.3 Phase Three

Let the allocation $\mathbf{X}$ and sequence σ be the outputs of the first two phases. After satisfying all our desired properties, we now allocate the remaining unallocated unit bundles to a third-party to obtain a full EFX allocation. Note that this is the only phase where we use the assumption that our input instance is triangle-free.

Algorithm 3 takes $(\mathbf{X}, \sigma)$ that satisfy properties 1-7 as input, and then for every agent j who envies some agent i, adds $B_i^{(1)}(\mathbf{X})$ to agent j. We prove that this algorithm outputs a complete EFX allocation in Lemma 3.13.

Algorithm 3: Dumping Remaining Items

Input: $(\mathbf{X}, \sigma)$ satisfying properties 1-7
Output: An EFX allocation
for *every envied agent* i **do**
  Let j be the agent who envies i.
  $X_j \leftarrow X_j \cup B_i^{(1)}(\mathbf{X})$
return X

Lemma 3.13. *Algorithm 3 terminates in at most* n *rounds, and outputs an EFX allocation.*

Proof. Algorithm 3 terminates in at most n iterations since there are fewer than n envied agents. Prior to the execution of Algorithm 3, Observation 3.10 implies that every unallocated unit bundle resides in one of the following cases:

- **Case 1:** Between an envied agent i and a non-envied agent j where agent j does not envy i. Here, agent j has received a unit bundle from $E(i, j)$ but agent i has not received a unit bundle from $E(i, j)$.
- **Case 2:** Between two envied agents i and k. Let j and l be the agents envying i and k, respectively. Note that since our input instance does not contain a cycle of length three, we have $j \neq l$.

We first argue that after execution of Algorithm 3, all of the items will be allocated, which is to show that all the unallocated unit bundles will be allocated. In case 1, $B_i^{(1)}(\mathbf{X})$ contains the unallocated unit bundle in $E(i, j)$, and it will be dumped to the agent who envies agent i. In case 2, since both of the unit bundles of $E(i, k)$ are unallocated, by definition, we get that $B_{i,k}^{(1)}(\mathbf{X}) = E(i, k) \setminus B_{k,i}^{(1)}(\mathbf{X})$. Therefore, since $B_{i,k}^{(1)}(\mathbf{X}) \subseteq B_i^{(1)}(\mathbf{X})$, and $B_{k,i}^{(1)}(\mathbf{X}) \subseteq B_k^{(1)}(\mathbf{X})$, unit bundles of $E(i, k)$ will be allocated to j and l.

Next, we show that allocation remains EFX. Suppose agent i is an arbitrary non-envied agent, and she envies agents $j_1, \ldots, j_z$ prior to execution of Algorithm 3. Since, allocation was EFX prior to the execution of Algorithm 3, and no envied agent receives any new item, and also no agent loses any item, we only need to show that no agent envies agent i after that $X'_i = X_i \cup B_{j_1}^{(1)}(\mathbf{X}) \cup \ldots \cup B_{j_z}^{(1)}(\mathbf{X})$ is allocated to i.

For any agent $j_p \in \{j_1, \ldots, j_z\}$, we have that

$$\bigcup_{q \in [z] \setminus \{p\}} B_{j_q}^{(1)}(\mathbf{X}) \cap E(j_p) = \emptyset$$

because the underlying multi-graph is triangle-free, and j_p and j_q are both adjacent to i, so j_p and j_q are not adjacent. Hence,

$$v_{j_p}(X_{j_p}) \geq v_{j_p}(X_i \cup B_{j_p}^{(1)}(\mathbf{X})) = v_{j_p}(X'_i),$$

where the first inequality is due to property 7.

Now, it remain to show the same for agents $r \notin \{i, j_1, \ldots, j_z\}$. Note that since the underlying multi-graph is triangle-free, r is either not adjacent to i, meaning $X_i \cap E(r) = \emptyset$ or r is not adjacent to every $j_q \in \{j_1, \ldots, j_z\}$, meaning $\bigcup_{q \in [z]} B_{j_q}^{(1)}(\mathbf{X}) \cap E(r) = \emptyset$. Thus, we only need to show that $v_r(X_r) \geq v_r(X_i)$ and $v_r(X_r) \geq v_r(\bigcup_{q \in [z]} B_{j_q}^{(1)}(\mathbf{X}))$. The first inequality holds since agent i was non-envied. For the second inequality, note that

$$\bigcup_{q \in [z]} B_{j_q}^{(1)}(\mathbf{X}) \cap E(r) = \bigcup_{q \in [z]} B_{j_q,r}^{(1)}(\mathbf{X}).$$

If r is non-envied, since every unit bundle $B_{j_q,r}^{(1)}(\mathbf{X})$ is in $U_r(\mathbf{X})$ then by monotonicity and property 6:

$$v_r(X_r) \geq v_r(U_r(\mathbf{X})) \geq v_r(\bigcup_{q \in [z]} B_{j_q,r}^{(1)}(\mathbf{X})) = v_r(\bigcup_{q \in [z]} B_{j_q}^{(1)}(\mathbf{X})).$$

If r is envied, since every j_q is also envied, both unit bundles between r and j_q are unallocated, so $B_{j_q,r}^{(1)}(\mathbf{X}) = B_{r,j_q}^{(2)}(\mathbf{X})$, and therefore,

$$\bigcup_{q \in [z]} B_{j_q}^{(1)}(\mathbf{X}) \cap E(r) = \bigcup_{q \in [z]} B_{j_q,r}^{(1)}(\mathbf{X}) = \bigcup_{q \in [z]} B_{r,j_q}^{(2)}(\mathbf{X}) \subseteq B_r^{(2)}(\mathbf{X}),$$

Thus, recalling the monotonicity of the valuation functions and by property 7, we find

$$v_r(X_r) \geq v_r(B_r^{(2)}(\mathbf{X})) \geq v_r(\bigcup_{q \in [z]} B_{j_q}^{(1)}(\mathbf{X}) \cap E(r)) = v_r(\bigcup_{q \in [z]} B_{j_q}^{(1)}(\mathbf{X})).$$

Either way, the desired inequality holds. Thus, the proof is complete. □

Now, we have proven our main result, i.e., we can obtain an EFX allocation for triangle-free multi-graphs if we do the following: first, execute Algorithm 1 for at most n times to satisfy properties 1-4. Then, execute Algorithms 2, and 3 in order.

3.4 Running Time

In this part, we analyze the running time of our algorithm. By Lemma 3.5, executing Algorithm 1 at most n times satisfies properties 1-4, where each execution takes polynomial time. It is also clear that Algorithms 2 and 3 terminate in polynomial time, as discussed in previous sections. Therefore, the only part of our algorithm that determines the running time is the computation of the *Cut Configurations* between any pair of vertices. Therefore, we briefly discuss the running time of computing such configurations for different classes of valuation functions.

EFX Cut Between Two Agents. It is well known that the simple algorithm introduced by Plaut and Roughgarden [49], later referred to as the PR algorithm by Akrami et al. [4], takes as input a monotone valuation function v, a set of goods S, and a natural number k, and outputs a partition $(Y_1, \ldots, Y_k)$ of S such that every bundle in the partition is EFX-feasible for an agent with valuation v. The PR algorithm runs in pseudo-polynomial time for monotone valuations (see the proof of Theorem 3.49 in [13]). Also, Ashuri et al. [13] proposed a modified version of the PR algorithm and showed that, for $k = 2$ and a cancelable valuation function, it runs in polynomial time (see Lemma 4.7 in [13]).

Combining the results above from [13], we can conclude that our result terminates in polynomial time for cancelable valuation functions, which are a strict generalization of additive valuations, and in pseudo-polynomial time for general monotone valuations.

4 CONCLUSION

In this work, we have studied a model that captures a setting where each item is relevant to at most two agents, while any pair of agents can have an arbitrary number of items that they both value, represented via a multi-graph. We advanced the understanding of fair allocation by resolving a key open question on the existence of EFX allocations in multi-graph valuation settings. By proving existence whenever the underlying graph contains no three-cycles, our work strictly generalizes prior results and significantly broadens the class of instances for which EFX fairness can be guaranteed. Beyond existence, we contribute algorithmic insights by establishing a pseudo-polynomial procedure for computing EFX allocations under monotone valuations, which becomes polynomial when valuations are cancelable. These results mark one of the few known cases in which EFX allocations exist for an arbitrary number of agents, thereby moving the boundary of tractable and guaranteed fairness further than previously known. Future directions include tightening the structural assumptions on the graph, that is, allowing cycles of length three to be presented in the graph structure, and proving the existence of EFX allocations. We conjecture that any fair division instance represented via a multi-graph admits an EFX allocation. Moreover, computing such an allocation might be possible by adding certain useful properties to our initial orientation (the output of our first phase). Another interesting direction is to explore trade-offs between fairness and efficiency, for example, by approximating Social Welfare or Nash Social Welfare.

Finally, we believe that the algorithmic insights gained by our work on triangle-free multi-graphs will contribute to future research on the existence and computation of EFX allocations in more general settings.

REFERENCES

- [1] Mahyar Afshinmehr, Matin Ansaripour, Alireza Danaei, and Kurt Mehlhorn. 2024. Approximate EFX and Exact tEFX Allocations for Indivisible Chores: Improved Algorithms. arXiv:2410.18655 [cs.GT] https://arxiv.org/abs/2410.18655
- [2] Mahyar Afshinmehr, Arash Ashuri, Pouria Mahmoudkhan, and Kurt Mehlhorn. 2025. EFX Allocations Exist on Triangle-Free Multi-Graphs. arXiv:2512.21644 [cs.GT] https://arxiv.org/abs/2512.21644
- [3] Mahyar Afshinmehr, Alireza Danaei, Mehrafarin Kazemi, Kurt Mehlhorn, and Nidhi Rathi. 2025. EFX Allocations and Orientations on Bipartite Multi-graphs: A Complete Picture. In *Proceedings of the 24th International Conference on Autonomous Agents and Multiagent Systems, AAMAS 2025, Detroit, MI, USA, May 19-23, 2025*. International Foundation for Autonomous Agents and Multiagent Systems / ACM.
- [4] Hannaneh Akrami, Noga Alon, Bhaskar Ray Chaudhury, Jugal Garg, Kurt Mehlhorn, and Ruta Mehta. 2023. 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*. ACM, 61.
- [5] Hannaneh Akrami, Ryoga Mahara, Kurt Mehlhorn, and Nidhi Rathi. 2026. Achieving EF1 and Epistemic EFX Guarantees Simultaneously. arXiv:2602.11732 [cs.GT] https://arxiv.org/abs/2602.11732
- [6] Hannaneh Akrami and Nidhi Rathi. 2025. Achieving Maximin Share and EFX/EF1 Guarantees Simultaneously. In *AAAI-25, Sponsored by the Association for the Advancement of Artificial Intelligence, February 25 - March 4, 2025, Philadelphia, PA, USA*. AAAI Press, 13529–13537.
- [7] Hannaneh Akrami and Nidhi Rathi. 2025. Epistemic EFX Allocations Exist for Monotone Valuations. In *AAAI-25, Sponsored by the Association for the Advancement of Artificial Intelligence, February 25 - March 4, 2025, Philadelphia, PA, USA*. AAAI Press.
- [8] Hannaneh Akrami, Rojin Rezvan, and Masoud Seddighin. 2022. An EF2X Allocation Protocol for Restricted Additive Valuations. In *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI 2022*. ijcai.org, 17–23.
- [9] Georgios Amanatidis, Haris Aziz, Georgios Birmpas, Aris Filos-Ratsikas, Bo Li, Hervé Moulin, Alexandros A. Voudouris, and Xiaowei Wu. 2023. Fair division of indivisible goods: Recent progress and open questions. *Artif. Intell.* 322 (2023), 103965.
- [10] Georgios Amanatidis, Aris Filos-Ratsikas, and Alkmini Sgouritsa. 2024. Pushing the Frontier on Approximate EFX Allocations. In *Proceedings of the 25th ACM Conference on Economics and Computation, EC 2024, New Haven, CT, USA, July 8-11, 2024*. ACM.
- [11] Georgios Amanatidis, Evangelos Markakis, and Apostolos Ntokos. 2020. Multiple Birds with One Stone: Beating 1/2 for EFX and GMMS via Envy Cycle Elimination. AAAI Press, 1790–1797.
- [12] Arash Ashuri and Vasilis Gkatzelis. 2025. Simultaneously Satisfying MXS and EFL. In *Proceedings of the 26th ACM Conference on Economics and Computation, EC 2025, Stanford University, Stanford, CA, USA, July 7-10, 2025*. ACM, 689–718.
- [13] Arash Ashuri, Vasilis Gkatzelis, and Alkmini Sgouritsa. 2025. EF2X Exists for Four Agents. In *AAAI-25, Sponsored by the Association for the Advancement of Artificial Intelligence, February 25 - March 4, 2025, Philadelphia, PA, USA*. AAAI Press.
- [14] Siddharth Barman, Arpita Biswas, Sanath Kumar Krishna Murthy, and Yadati Narahari. 2018. Groupwise Maximin Fair Allocation of Indivisible Goods. AAAI Press, 917–924.
- [15] Siddharth Barman, Debajyoti Kar, and Shraddha Pathak. 2024. Parameterized Guarantees for Almost Envy-Free Allocations. In *Proceedings of the 23rd International Conference on Autonomous Agents and Multiagent Systems, AAMAS 2024, Auckland, New Zealand, May 6-10, 2024*. International Foundation for Autonomous Agents and Multiagent Systems / ACM.
- [16] Benjamin Aram Berendsohn, Simona Boyadzhiyska, and László Kozma. 2022. Fixed-Point Cycles and Approximate EFX Allocations. In *47th International Symposium on Mathematical Foundations of Computer Science, MFCS 2022 (LIPIcs, Vol. 241)*. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 17:1–17:13.
- [17] Ben Berger, Avi Cohen, Michal Feldman, and Amos Fiat. 2022. Almost Full EFX Exists for Four Agents. AAAI Press, 4826–4833.
- [18] Umang Bhaskar and Yeshwant Pandit. 2024. EFX Allocations on Some Multigraph Classes. *CoRR* abs/2412.06513 (2024).
- [19] Václav Blazej, Sushmita Gupta, M. S. Ramanujan, and Peter Strulo. 2025. Tractable Graph Structures in EFX Orientation. In *Algorithmic Game Theory - 18th International Symposium, SAGT 2025, Proceedings (Lecture Notes in Computer Science, Vol. 15953)*. Springer.
- [20] Steven J Brams and Alan D Taylor. 1996. *Fair Division: From cake-cutting to dispute resolution*. Cambridge University Press.
- [21] Felix Brandt and Ariel D Procaccia. 2016. *Handbook of computational social choice*. Cambridge University Press.
- [22] Xiaolin Bu, Jiaxin Song, and Ziqi Yu. 2023. EFX Allocations Exist for Binary Valuations. In *Frontiers of Algorithmics - 17th International Joint Conference, IJTCS-FAW 2023 Macau, China, August 14-18, 2023, Proceedings (Lecture Notes in Computer Science, Vol. 13933)*. Springer.
- [23] Eric Budish. 2010. The combinatorial assignment problem: approximate competitive equilibrium from equal incomes, Moshe Dror and Greys Sosic (Eds.). ACM, 74:1.
- [24] Eric Budish and Estelle Cantillon. 2012. The multi-unit assignment problem: Theory and evidence from course allocation at Harvard. *American Economic Review* 102, 5 (2012), 2237–2271.
- [25] Ioannis Caragiannis, Jugal Garg, Nidhi Rathi, Eklavya Sharma, and Giovanna Varricchio. 2023. New Fairness Concepts for Allocating Indivisible Items. ijcai.org, 2554–2562.
- [26] Ioannis Caragiannis, Nick Gravin, and Xin Huang. 2019. Envy-Freeness Up to Any Item with High Nash Welfare: The Virtue of Donating Items. In *Proceedings of the 2019 ACM Conference on Economics and Computation, EC 2019*, Anna R. Karlin, Nicole Immorlica, and Ramesh Johari (Eds.). ACM, 527–545.
- [27] Ioannis Caragiannis, David Kurokawa, Hervé Moulin, Ariel D. Procaccia, Nisarg Shah, and Junxing Wang. 2019. The Unreasonable Fairness of Maximum Nash Welfare. *ACM Trans. Economics and Comput.* 7, 3 (2019), 12:1–12:32.
- [28] Bhaskar Ray Chaudhury, Jugal Garg, and Kurt Mehlhorn. 2024. EFX Exists for Three Agents. *J. ACM* 71, 1 (2024), 4:1–4:27.
- [29] Bhaskar Ray Chaudhury, Jugal Garg, Kurt Mehlhorn, Ruta Mehta, and Pranabendu Misra. 2021. Improving EFX Guarantees through Rainbow Cycle Number. In *EC '21: The 22nd ACM Conference on Economics and Computation*. ACM, 310–311.
- [30] Bhaskar Ray Chaudhury, Telikepalli Kavitha, Kurt Mehlhorn, and Alkmini Sgouritsa. 2021. A Little Charity Guarantees Almost Envy-Freeness. *SIAM J. Comput.* 50, 4 (2021), 1336–1358.
- [31] George Christodoulou, Amos Fiat, Elias Koutsoupias, and Alkmini Sgouritsa. 2023. Fair allocation in graphs. In *Proceedings of the 24th ACM Conference on Economics and Computation, EC 2023, London, United Kingdom, July 9-12, 2023*. ACM, 473–488.
- [32] Vasilis Christoforidis and Christodoulos Santorinaios. 2024. On the Pursuit of EFX for Chores: Non-existence and Approximations. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence, IJCAI-24*, Kate Larson (Ed.). International Joint Conferences on Artificial Intelligence Organization, 2713–2721. Main Track.
- [33] Argyrios Deligkas, Eduard Eiben, Tiger-Lily Goldsmith, and Viktoriia Korchemna. 2025. EF1 and EFX Orientations. In *Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence, IJCAI 2025, Montreal, Canada, August 16-22, 2025*. ijcai.org.
- [34] Raul Etkin, Abhay Parekh, and David Tse. 2007. Spectrum sharing for unlicensed bands. *IEEE Journal on Selected Areas in Communications* 25, 3 (2007), 517–528.
- [35] Alireza Farhadi, Mohammad Taghi Hajiaghayi, Mohamad Latifian, Masoud Seddighin, and Hadi Yami. 2021. Almost Envy-freeness, Envy-rank, and Nash Social Welfare Matchings. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI*. AAAI Press, 5355–5362.
- [36] Uriel Feige. 2025. The residual maximin share. *CoRR* abs/2505.19961 (2025). https://doi.org/10.48550/ARXIV.2505.19961 arXiv:2505.19961
- [37] Jugal Garg and Aniket Murhekar. 2025. Existence of 2-EFX Allocations of Chores. arXiv:2507.19461 [cs.GT] https://arxiv.org/abs/2507.19461
- [38] Jugal Garg, Aniket Murhekar, and John Qin. 2025. Constant-Factor EFX Exists for Chores *(STOC '25)*. Association for Computing Machinery.
- [39] Vishwa Prakash HV, Pratik Ghosal, Prajakta Nimbhorkar, and Nithin Varma. 2025. EFX Exists for Three Types of Agents. In *Proceedings of the 26th ACM Conference on Economics and Computation, EC 2025, Stanford University, Stanford, CA, USA, July 7-10, 2025*. ACM, 101–128.
- [40] Vishwa Prakash HV, Ruta Mehta, and Prajakta Nimbhorkar. 2025. Almost and Approximate EFX for Few Types of Agents. *CoRR* abs/2508.15380 (2025).
- [41] Shayan Chashm Jahan, Masoud Seddighin, Seyed Mohammad Seyed Javadi, and Mohammad Sharifi. 2023. Rainbow Cycle Number and EFX Allocations: (Almost) Closing the Gap. In *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence, IJCAI 2023*. ijcai.org, 2572–2580.
- [42] Alireza Kaviani, Alireza Keshavarz, Masoud Seddighin, and AmirMohammad Shahrezaei. 2025. Improved Approximate EFX Guarantees for Multigraphs. *CoRR* abs/2506.09288 (2025). https://doi.org/10.48550/ARXIV.2506.09288 arXiv:2506.09288
- [43] Alireza Kaviani, Masoud Seddighin, and AmirMohammad Shahrezaei. 2024. Almost Envy-free Allocation of Indivisible Goods: A Tale of Two Valuations. arXiv:2407.05139 [cs.GT]
- [44] Richard J. Lipton, Evangelos Markakis, Elchanan Mossel, and Amin Saberi. 2004. On approximately fair allocations of indivisible goods. In *Proceedings 5th ACM Conference on Electronic Commerce (EC-2004)*. ACM, 125–131.
- [45] Ryoga Mahara. 2024. Extension of Additive Valuations to General Valuations on the Existence of EFX. *Math. Oper. Res.* 49, 2 (2024), 1263–1277.
- [46] Ryoga Mahara. 2025. Existence of Fair and Efficient Allocation of Indivisible Chores. arXiv:2507.09544 [cs.GT] https://arxiv.org/abs/2507.09544
- [47] Evangelos Markakis and Christodoulos Santorinaios. 2023. Improved EFX Approximation Guarantees under Ordinal-based Assumptions. In *Proceedings of the 2023 International Conference on Autonomous Agents and Multiagent Systems*. ACM, 591–599.

[48] Hervé Moulin. 2004. *Fair Division and Collective Welfare.* MIT press.
[49] Benjamin Plaut and Tim Roughgarden. 2020. Almost Envy-Freeness with General Valuations. *SIAM J. Discret. Math.* 34, 2 (2020), 1039–1068.
[50] John Winsor Pratt and Richard Jay Zeckhauser. 1990. The fair and efficient division of the Winsor family silver. *Management Science* 36, 11 (1990), 1293–1301.
[51] Jack Robertson and William Webb. 1998. *Cake-cutting algorithms: Be fair if you can.* AK Peters/CRC Press.
[52] Alkmini Sgouritsa and Minas Marios Sotiriou. 2025. On the Existence of EFX Allocations in Multigraphs. In *Proceedings of the 24th International Conference on Autonomous Agents and Multiagent Systems, AAMAS 2025, Detroit, MI, USA, May 19-23, 2025.* International Foundation for Autonomous Agents and Multiagent Systems / ACM.
[53] Hugo Steinhaus. 1949. Sur la division pragmatique. *Econometrica* 17 (1949), 315–319.
[54] Thomas Vossen. 2002. *Fair Allocation Concepts in Air Traffic Management.* Ph.D. Dissertation. University of Maryland.
[55] Jinghan A. Zeng and Ruta Mehta. 2025. On the Structure of EFX Orientations on Graphs. In *Proceedings of the 24th International Conference on Autonomous Agents and Multiagent Systems, AAMAS 2025, Detroit, MI, USA, May 19-23, 2025.* International Foundation for Autonomous Agents and Multiagent Systems / ACM.
[56] Shengwei Zhou and Xiaowei Wu. 2024. Approximately EFX allocations for indivisible chores. *Artif. Intell.* 326, C (Jan. 2024), 21. https://doi.org/10.1016/j.artint.2023.104037
[57] Yu Zhou, Tianze Wei, Minming Li, and Bo Li. 2024. 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, IJCAI 2024, Jeju, South Korea, August 3-9, 2024.* ijcai.org.