Pareto-Optimal Allocation of Indivisible Goods with Connectivity Constraints

Ayumi Igarashi
Kyushu University
Fukuoka, Japan
igarashi@agent.inf.kyushu-u.ac.jp

Dominik Peters
University of Oxford
Oxford, U.K.
dominik.peters@cs.ox.ac.uk

Abstract

We study the problem of allocating indivisible items to agents with additive valuations, under the additional constraint that bundles must be connected in an underlying item graph. Previous work has considered the existence and complexity of fair allocations. We study the problem of finding an allocation that is Pareto-optimal. While it is easy to find an efficient allocation when the underlying graph is a path or a star, the problem is NP-hard for many other graph topologies, even for trees of bounded pathwidth or of maximum degree 3. We show that on a path, there are instances where no Pareto-optimal allocation satisfies envy-freeness up to one good, and that it is NP-hard to decide whether such an allocation exists, even for binary valuations. We also show that, for a path, it is NP-hard to find a Pareto-optimal allocation that satisfies maximin share, but show that a moving-knife algorithm can find such an allocation when agents have binary valuations that have a non-nested interval structure.

1 Introduction

In mechanism design, Pareto-optimality is a basic desideratum: if we select an outcome that is Pareto-dominated by another, users will justifiably complain. In simple settings, it is computationally trivial to find a Pareto-optimum (e.g., via serial dictatorship). Thus, it is usually sought to be satisfied together with other criteria (like fairness or welfare maximisation). However, in more complicated settings, even Pareto-optimality may be elusive.

We study the classical problem of allocating indivisible items among agents who have (typically additive) preferences over bundles. Following a recent model of Bouveret et al. (2017), we are interested in settings where the set of items has additional structure specified by a graph G over the items. Agents are only interested in receiving a bundle of items that is connected in G. This model is particularly relevant when the items have a spatial or temporal structure, for example, if we wish to allocate land, rooms, or time slots to agents. Time slots, for instance, are naturally ordered in a sequence, and agents will often only value being allocated a contiguous chunk of time, particularly when restart costs are prohibitive.

Given agents' preferences over (connected) bundles, we wish to find an allocation that is *Pareto-optimal* (or *Pareto-*

efficient), that is, a connected allocation such that there is no other *connected* allocation which makes some agent strictly better off while making no agent worse off. Now, in the standard setting without connectivity constraints and with additive valuations, it is straightforward to find Pareto-optima: For example, we can allocate each item to a person who has the highest valuation for it (maximizing utilitarian social welfare in the process), or we can run a serial dictatorship. Neither of these approaches respects connectivity constraints. In fact, we show that it is NP-hard to construct a Pareto-optimal allocation under connectivity constraints, unless G is extremely simple.

Recent work on the allocation of indivisible items has focussed particularly on ensuring *fairness*. Two well-studied fairness notions are due to Budish (2011), who introduced the *maximin fair share* (MMS) and *envy-freeness up to one good* (EF1). Both concepts have natural analogues in the setting with connectivity constraints (Bouveret et al. 2017; Bilò et al. 2018). An important question is whether there is a tradeoff between efficiency and fairness, or whether both are simultaneously achievable. Without connectivity constraints, these notions tend to be compatible: For example, with additive valuations, the maximum Nash welfare solution satisfies EF1 and is also Pareto-optimal (Caragiannis et al. 2016). We investigate these tradeoffs in the connected setting.

Contributions.

- For additive valuations, we show that one can find a Pareto-optimum in polynomial time when G is a path or a star.

- We show that, unless P = NP, there is no polynomial-time algorithm that finds a Pareto-optimum when G is a tree, even if valuations are additive and binary, and even if the tree has bounded pathwidth, bounded diameter, or bounded maximum degree. Finding a Pareto-optimum is also hard when valuations are 2-additive and G is a path or a star.

- When G is a tree, there always exists an allocation which is both Pareto-optimal and satisfies MMS. However, such an allocation is NP-hard to find, even when G is a path; the problem stays hard when weakening MMS to α-MMS for any $\alpha > 0$. For a restricted class of binary valuations (non-nested intervals), we give a polynomial-time algorithm.

- When G is a path, we give examples with binary additive valuations for which no Pareto-optimal EF1 allocation exists, and show that it is NP-hard to decide existence.

	general	complete	tree	path
PO	NP-hard*	poly-time	NP-hard*	poly-time
PO & MMS	NP-hard*		NP-hard*	NP-hard*
PO & EF1	NP-hard	poly-time†	NP-hard	NP-hard

Table 1: Overview of our complexity results. Hardness results marked * hold under Turing reductions. The result † refers to a pseudo-polynomial algorithm by Barman, Murthy, and Vaish (2018). Our hardness results hold even for additive and binary valuations.

Related Work. There is a rich body of the literature on fair division of a *divisible* cake into connected pieces. Such a division satisfying envy-freeness always exists (Stromquist 1980); nevertheless, it cannot be obtained in finite steps even when the cake is divided among three agents (Stromquist 2008). In contrast, several efficient algorithms are known to yield a contiguous proportional allocation; see the survey by Lindner and Rothe (2016) for more details.

The relation between efficiency and fairness with connected pieces is also well-understood for divisible items. Aumann and Dombb (2015) studied the utilitarian social welfare of fair allocations under connectivity constraints. The papers by Bei et al. (2012) and Aumann, Dombb, and Hassidim (2013) considered the computational complexity of finding an allocation with connected pieces maximizing utilitarian social welfare. Bei et al. (2012) showed that utilitarian social welfare is inapproximable when requiring that the allocation satisfy proportionality; however, without the proportionality requirement, Aumann, Dombb, and Hassidim (2013) proved that there is a polynomial-time constant-factor approximation algorithm for finding an allocation maximizing utilitarian social welfare. The algorithm by Aumann, Dombb, and Hassidim (2013) works also for indivisible items and so applies to our setting when G is a path. A paper by Conitzer, Derryberry, and Sandholm (2004) considers combinatorial auctions; translated to our setting, their results imply that one can find a Pareto-optimal connected allocation in polynomial time, when G is a graph of bounded treewidth and agents have *unit demand*: each agent specifies a connected demanded bundle such that agents have positive utility if and only if they obtain a superset of the demanded bundle.

In the context of division of indivisible items, Bouveret et al. (2017) formalized the model of fair division with the extra feature that each bundle needs to be connected in the underlying item graph. While they showed that finding a connected allocation that is envy-free or proportional is NP-hard even on paths, they proved that an allocation satisfying the maximin fair share always exists and can be found in polynomial time when the graph is acyclic; subsequently Lonc and Truszczynski (2018) studied the computational complexity of finding an MMS allocation when the graph G is a cycle. Independently of Bouveret et al. (2017), Suksompong (2017) considered the problem when the items lie on a path, obtaining approximations to several fairness notions such as envy-freeness and proportionality. The recent works of Bilò et al. (2018) and Oh, Procaccia, and Suksompong (2019)

study the existence of EF1 allocations with connected pieces. They both showed that an EF1 allocation exists when agents have identical valuations. Bilò et al. (2018) also proved that for up to four agents with arbitrary monotonic valuations, an EF1 allocation connected on a path is guaranteed to exist.

With no connectivity constraints, Aziz et al. (2016) studied the computational complexity of finding Pareto-improvements of a given allocation when agents have additive preferences. Technically, our hardness proofs use similar techniques to hardness proofs obtained by Aziz, Brandt, and Harrenstein (2013) in the context of hedonic games.

Full version. A full version is available on arXiv (Igarashi and Peters 2018). It contains the proofs of Theorems 5, 6, 7, and 9, which are omitted here due to space constraints.

2 Preliminaries

For an integer $s \in \mathbb{N}$, write $[s] = \{1, 2, \ldots, s\}$. Let $N = [n]$ be a set of *agents* and $G = (V, E)$ be an undirected graph whose vertices are called *items*. A subset X of V is *connected* if it induces a connected subgraph of G. We write $\mathcal{C}(V) \subseteq 2^V$ for the set of connected subsets of V, also called *bundles*.

Each agent $i \in N$ has a *valuation function* $u_i : \mathcal{C}(V) \to \mathbb{R}$ over connected bundles which satisfies $u_i(\emptyset) = 0$ and is *monotonic*, so $X \subseteq Y$ implies $u_i(X) \leqslant u_i(Y)$. A valuation function u_i is *additive* if $u_i(X) = \sum_{v \in X} u_i(\{v\})$ for each $X \in \mathcal{C}(V)$. We write $u_i(v) = u_i(\{v\})$ for short. An additive valuation function is *binary* if $u_i(v) \in \{0, 1\}$ for all $v \in V$. If an agent i has a binary valuation function, we say that i *approves* item v if $u_i(v) = 1$.

A (connected) *allocation* is a function $\pi : N \to \mathcal{C}(V)$ assigning each agent a connected bundle of items, such that each item is allocated exactly once, i.e., $\bigcup_{i \in N} \pi(i) = V$ and $\pi(i) \cap \pi(j) = \emptyset$ for each pair of distinct agents $i, j \in N$. For an allocation π and a subset N' of agents, we denote by $\pi|_{N'}$ the allocation restricted to N'.

Given an allocation π, another allocation π' is a *Pareto-improvement* of π if $u_i(\pi'(i)) \geqslant u_i(\pi(i))$ for all $i \in N$ and $u_j(\pi'(j)) > u_j(\pi(j))$ for some $j \in N$. We say that a connected allocation π is *Pareto-optimal* (or *Pareto-efficient*, or *PO* for short) if there is no *connected* allocation that is a Pareto-improvement of π. The *utilitarian social welfare* of an allocation π is $\sum_{i \in N} u_i(\pi(i))$. It is easy to see that a connected allocation which maximizes utilitarian social welfare among connected allocations is Pareto-optimal.

A connected allocation satisfies *EF1* (Bilò et al. 2018; Oh, Procaccia, and Suksompong 2019) if for any pair of agents $i, j \in N$, either $u_i(\pi(i)) \geqslant u_i(\pi(j))$ or there is an item $v \in \pi(j)$ such that $\pi(j) \setminus \{v\}$ remains connected and $u_i(\pi(i)) \geqslant u_i(\pi(j) \setminus \{v\})$. Thus, whenever i envies the bundle of agent j, then the envy vanishes if we remove one outer item from the envied bundle.

Let $\Pi_n(G)$ be the set of partitions of V into n connected bundles. The *maximin fair share* of an agent $i \in N$ is

$$\text{mms}_i = \max_{(P_1, \ldots, P_n) \in \Pi_n(G)} \min_{j \in [n]} u_i(P_j).$$

A connected allocation π is an *MMS allocation* if $u_i(\pi(i)) \geqslant \text{mms}_i$ for each agent $i \in N$. Bouveret et al. (2017) show that

if G is a tree, an MMS allocation exists. Note that this definition of the MMS value varies with the graph G, and may be lower than the standard MMS values where the maximisation is taken over all partitions, with no connectivity constraints.

Some graph-theoretic terminology: Given a graph $G = (V, E)$ and a subset $X \subseteq V$ of vertices, we denote by $G \setminus X$ the subgraph of G induced by $V \setminus X$. The *diameter* of G is the maximum distance between any pair of vertices.

3 Finding Some Pareto-Optimal Allocation

We start by considering the problem of producing *some* Pareto-optimal allocation, without imposing any additional constraints on the quality of this allocation. When there are no connectivity requirements (equivalently, when G is a complete graph) and valuations are additive, this problem is trivial: Simply allocate each item v separately to an agent i who has a highest valuation $u_i(v)$ for v. The resulting allocation maximizes utilitarian social welfare and is thus Pareto-optimal. When G is not complete, this procedure can produce disconnected bundles. We could try to give all items to a single agent (if the graph G is connected), but the result need not be Pareto-optimal if that agent has zero value for some items. Is it still possible to find a Pareto-optimal allocation for specific graph topologies in polynomial time?

Paths and Stars

For very simple graphs and additive valuations, the answer is positive. Our first algorithm works when G is a path. The algorithm identifies an agent i with a nonzero valuation for the item at the left end of the path G, and then allocates all items to i, except for any items at the right end of the path which i values at 0. We then recursively call the same algorithm to decide on how to allocate the remaining items.

Theorem 1. *When G is a path, and with additive valuations, a Pareto-optimal allocation can be found in polynomial time.*

Proof. The path G is given by $V = \{v_1, v_2, \ldots, v_m\}$ where $\{v_j, v_{j+1}\} \in E$ for each $j \in [m-1]$. For a subset V' of V, we denote by $\min V'$ the vertex of minimum index in V'.

We design a recursive algorithm $\mathcal{A}$ that takes as input a subset N' of agents, a subpath $G' = (V', E')$ of G, and a valuation profile $(u_i)_{i \in N'}$, and returns a Pareto-optimal allocation of the items in V' to the agents in N'. Without loss of generality, we may assume that there is an agent who likes the left-most vertex of G', i.e., $u_i(\min V') > 0$ for some $i \in N'$, since otherwise we can arbitrarily allocate that item later without affecting Pareto-optimality.

If $|N'| = 1$, then the algorithm allocates all items V' to the single agent. Suppose that $|N'| > 1$. The algorithm first finds an agent i who has positive value for $\min V'$; it then allocates to i a minimal connected bundle $V_i \subseteq V'$ containing all items in V' to which i assigns positive utility (so that $u_i(V_i) = u_i(V')$). To decide on the allocation of the remaining items, we apply $\mathcal{A}$ to the reduced instance $(N' \setminus \{i\}, G' \setminus V_i, (u_{i'})_{i' \in N' \setminus \{i\}})$.

We will prove by induction on $|N'|$ that the allocation produced by $\mathcal{A}$ is Pareto-optimal. This is clearly true when $|N'| = 1$. Suppose that $\mathcal{A}$ returns a Pareto-optimal allocation when $|N'| = k - 1$; we will prove it for $|N'| = k$. Let π be the allocation returned by $\mathcal{A}$, where $\mathcal{A}$ chose agent i and allocated the bundle V_i before making a recursive call. Assume for a contradiction that there is a Pareto-improvement π' of π. Thus, in particular, $u_i(\pi'(i)) \geqslant u_i(\pi(i))$. By the algorithm's choice of the bundle V_i, we must have $V_i \subseteq \pi'(i)$ and $u_i(\pi'(i)) = u_i(\pi(i))$. Thus, there is an agent j' different from i who receives strictly higher value in π' than in π.

Now, since $G \setminus \pi'(i)$ is a subgraph of $G \setminus V_i$, the allocation $\pi'|_{N' \setminus \{i\}}$ is an allocation for the instance $I' = (N' \setminus \{i\}, G' \setminus V_i, (u_{i'})_{i' \in N' \setminus \{i\}})$. Also, we have

- $u_j(\pi'(j)) \geqslant u_j(\pi(j))$ for all agents $j \in N' \setminus \{i\}$; and
- $u_{j'}(\pi'(j')) > u_{j'}(\pi(j'))$ for some $j' \in N' \setminus \{i\}$.

Thus, $\pi'|_{N' \setminus \{i\}}$ is a Pareto-improvement of the allocation $\pi|_{N' \setminus \{i\}}$. But $\pi|_{N' \setminus \{i\}}$ is the allocation returned by $\mathcal{A}$ for the instance I', contradicting the inductive hypothesis that $\mathcal{A}$ returns Pareto-optimal allocations for $|N'| = k - 1$. □

Another graph type for which we can find a Pareto-optimum is a star. In fact, we can efficiently calculate an allocation maximizing utilitarian social welfare. Note that when G is a star, at most one agent can receive two or more items. This allows us to translate welfare maximization into a bipartite matching instance.

Theorem 2. *When G is a star, and valuations are additive, a Pareto-optimal allocation can be found in polynomial time.*

Proof. We give an algorithm to find an allocation that maximizes the utilitarian social welfare. Let G be a star with center vertex c and $m - 1$ leaves. We start by guessing an agent $i \in N$ who receives the item c. By connectedness, every other agent can receive at most one (leaf) item. To allocate the leaf items, we construct a weighted bipartite graph H_i with bipartition $(N', V \setminus \{c\})$ where N' consists of agents $j \in N \setminus \{i\}$ together with $m - 1$ copies $i_1, i_2, \ldots, i_{m-1}$ of agent i. (These copies represent 'slots' in i's bundle.) Each edge of form $\{j, v\}$ for $j \in N \setminus \{i\}$ has weight $u_j(v)$ and each edge of form $\{i_k, v\}$ has weight $u_i(v)$.

Observe that each connected allocation in which i obtains c can be identified with a matching in H_i: Every leaf object is either matched with the agent receiving it, or is matched with some copy i_k of i if the object is part of i's bundle. Note that utilitarian social welfare of this allocation equals the total weight of the matching. Since one can find a maximum-weight matching in a bipartite graph in polynomial time (see, e.g., Korte and Vygen 2006), we can find an allocation of maximum utilitarian social welfare efficiently. □

We have shown that finding a Pareto-optimum is easy for paths and for stars. An interesting open problem is whether the problem is also easy for caterpillars, a class of graphs containing both paths and stars. One might be able to combine the approaches of Theorems 1 and 2 to handle them, but the details are difficult. Note that caterplliars are precisely the graphs of pathwidth 1; we discuss a negative result about graphs of pathwidth 2 below. Another open problem is whether finding a Pareto-optimum is easy when G is a cycle.

Hardness Results

For more general classes of graphs, the news is less positive. We show that, unless P = NP, there is no polynomial-time algorithm which produces a Pareto-optimal allocation when G is an arbitrary tree even for binary valuations. Notably, this result implies that it is NP-hard to find allocations maximizing any type of social welfare (utilitarian, leximin, Nash) when G is a tree, since such allocations are also Pareto-optimal.

To obtain our hardness result, we first consider a more general problem which is easier to analyze, namely the case where G is a forest. Since a Pareto-optimum always exists, we cannot employ the standard approach of showing that a decision problem is NP-hard via many-one reductions. Instead, we show (by a Turing reduction) that a polynomial-time algorithm producing a connected Pareto-optimal allocation could be used to solve an NP-complete problem in polynomial time.

Theorem 3. *Unless* P = NP, *there is no polynomial-time algorithm which finds a Pareto-optimal connected allocation when G is a union of vertex-disjoint paths of size 3, even if valuations are binary and additive.*

Proof. We give a Turing reduction from EXACT-3-COVER (X3C). Recall that an instance of X3C is given by a set of elements $X = \{x_1, x_2, \ldots, x_{3r}\}$ and a family $\mathcal{S} = \{S_1, \ldots, S_s\}$ of three-element subsets of X; it is a 'yes'-instance if and only if there is an *exact cover* $\mathcal{S}' \subseteq \mathcal{S}$ with $|\mathcal{S}'| = r$ and $\bigcup_{S \in \mathcal{S}'} S = X$. For a set $S \in \mathcal{S}$, order the three elements of S in some canonical way (e.g., alphabetically) and write S^1, S^2, S^3 for the elements in that order.

Given an instance $(X, \mathcal{S})$ of X3C, for each $S \in \mathcal{S}$, construct a path P_S on three vertices $v_{S,1}, v_{S,2}, v_{S,3}$. Let $G = \bigcup_{S \in \mathcal{S}} P_S$. For each element $x \in X$, we introduce an agent i_x with $u_{i_x}(v_{S,j}) = 1$ iff $S^j = x$, and $u_{i_x}(v_{S,j}) = 0$ otherwise. Thus, agent i_x approves all instances of x. We also introduce $s - r$ dummy agents $d_1, \ldots, d_{s-r}$ who approve every item, so $u_{d_k}(v_{S,j}) = 1$ for all j, k, S. Note that for each agent i_x, a highest-value connected bundle has value 1, while for a dummy agent d_k, a highest-value connected bundle has value 3.

Suppose we had an algorithm $\mathcal{A}$ which finds a Pareto-optimal allocation. We show how to use $\mathcal{A}$ to solve X3C. Run $\mathcal{A}$ on the allocation problem constructed above to obtain a Pareto-optimal allocation π. We claim that the X3C instance $(X, \mathcal{S})$ has a solution iff

$$u_{i_x}(\pi(i_x)) = 1 \text{ for all } x \in X \text{ and} \atop u_{d_k}(\pi(d_k)) = 3 \text{ for all } k \in [s-r]. \quad (3.1)$$

Since (3.1) is easy to check, this equivalence implies that $\mathcal{A}$ can be used to solve X3C, and hence our problem is NP-hard.

Suppose π satisfies (3.1). We construct a solution to the X3C instance. For each $k \in [s-r]$, since $u_{d_k}(\pi(d_k)) = 3$, we must have $\pi(d_k) = P_S$ for some $S \in \mathcal{S}$. Let $\mathcal{S}' = \{S \in \mathcal{S} : \pi(d_k) \neq P_S \text{ for all } k \in [s-r]\}$. Then $\mathcal{S}'$ is a solution: Clearly $|\mathcal{S}'| = r$; further, for every $x \in X$, we have that $\pi(i_x) \in P_S$ for some $S \in \mathcal{S}$, and since $u_{i_x}(\pi(i_x)) = 1$ by (3.1), this implies that $x \in S$. Thus, $\bigcup_{S \in \mathcal{S}'} S = X$. Hence, $\mathcal{S}'$ is a solution to the X3C instance $(X, \mathcal{S})$.

Conversely, suppose there is a solution $\mathcal{S}'$ to the instance of X3C, but suppose for a contradiction that π fails condition (3.1). Define the following allocation π^*: For each $x \in X$, identify a set $S \in \mathcal{S}'$ and an index $j \in [3]$ such that $S^j = x$ and set $\pi^*(i_x) = \{v_{S,j}\}$; next, write $\mathcal{S} \setminus \mathcal{S}' = \{S'_1, \ldots, S'_{s-r}\}$ and set $\pi^*(d_k) = \{v_{S'_k,1}, v_{S'_k,2}, v_{S'_k,3}\}$ for each $k \in [s-r]$. Then π^* satisfies (3.1). Since π fails (3.1), the allocation π^* Pareto-dominates π, contradicting that π is Pareto-optimal. Hence, π satisfies (3.1), as desired. $\square$

Building on this reduction, we find that it is also hard to find a Pareto-efficient allocation if G is a tree (rather than a forest).

Theorem 4. *Unless* P = NP, *there is no polynomial-time algorithm which finds a Pareto-optimal connected allocation when G is a tree, even if valuations are binary and additive.*

Proof. To extend the reduction in the proof of Theorem 3 to trees, we first 'double' the reduction, by making a copy of each object and a copy of each agent with the same preference as the original agent. Specifically, given an instance $(X, \mathcal{S})$ of X3C, we create the same instance as in the proof of Theorem 3; that is, we make a path $P_S = (v_{S,1}, v_{S,2}, v_{S,3})$ for each $S \in \mathcal{S}$, and construct agent i_x for each $x \in X$ and dummy agents $d_1, d_2, \ldots, d_{s-r}$ with the same binary valuations.

In addition, we make a path $\hat{P}_S$ of copies $\hat{v}_{S,1}, \hat{v}_{S,2}, \hat{v}_{S,3}$ of each $S \in \mathcal{S}$. We then make a copy $\hat{i}_x$ of each agent i_x $(x \in X)$ together with copies $\hat{d}_1, \hat{d}_2, \ldots, \hat{d}_{s-r}$ of the dummy agents. We also introduce a new item c which serves as the center of a tree; specifically, we attach the center to the middle vertex $v_{S,2}$ of the path P_S, and the middle vertex $\hat{v}_{S,2}$ of the path $\hat{P}_S$, for each $S \in \mathcal{S}$. The resulting graph G is a tree consisting of $2r + 2|\mathcal{S}|$ paths of length 3, each attached to the vertex c by their middle vertex. See Figure 1.

No agent has positive value for the center item c. Copied agents only value copied objects and have the same valuations as the corresponding original agents, and non-copied agents only value non-copied objects. Formally, for each element $x \in X$, each $k \in [s-r]$, and each item v, agents' binary valuations are given as follows:

- $u_{i_x}(v) = 1$ iff $v = v_{S,j}$ and $S^j = x$;
- $u_{d_k}(v) = 1$ iff $v = v_{S,j}$ for some S, j;
- $u_{\hat{i}_x}(v) = 1$ iff $v = \hat{v}_{S,j}$ and $S^j = x$;
- $u_{\hat{d}_k}(v) = 1$ iff $v = \hat{v}_{S,j}$ for some S, j.

Write $N_o = \{i_x : x \in X\} \cup \{d_1, d_2, \ldots, d_{s-r}\}$ for the set of original agents, and $V_o = \bigcup_{S \in \mathcal{S}} \{v_{S,1}, v_{S,2}, v_{S,3}\}$ for the set of original items.

Suppose we had an algorithm $\mathcal{A}$ which finds a Pareto-optimal allocation. We show how to use $\mathcal{A}$ to solve X3C. Run $\mathcal{A}$ on the allocation problem constructed above to obtain a Pareto-optimum π. We may suppose without loss of generality that $c \notin \pi(i)$ for any $i \in N_o$, since otherwise we can swap the roles of the originals and the copies. We may further assume that each original agent $i \in N_o$ only receives original items, i.e., $\pi(i) \subseteq V_o$, since we can move any other items from $\pi(i)$ into other bundles without making anyone worse off. Hence, since $c \notin \pi(i)$, we see that $\pi(i) \subseteq P_S$ for
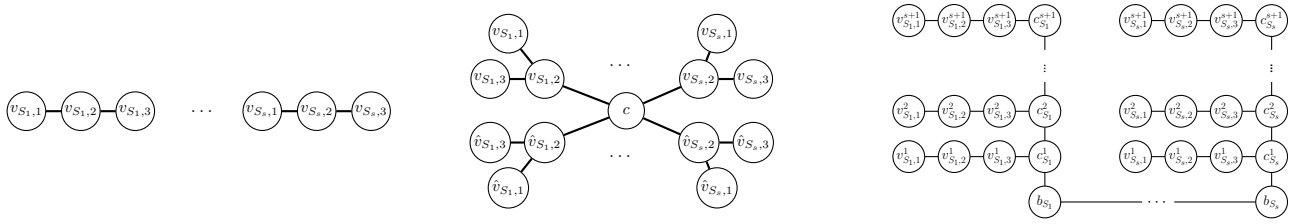
Figure 1: Graphs constructed in the proofs of Theorem 3, Theorem 4, and Theorem 5 (pictured left-to-right).

some $S \in \mathcal{S}$ because $\pi(i)$ is connected in G. This shows that $u_{i_x}(\pi(i_x)) \leqslant 1$ for all $x \in X$ and $u_{d_k}(\pi(d_k)) \leqslant 3$ for all $k \in [s-r]$. We prove that the X3C instance has a solution iff

$$u_{i_x}(\pi(i_x)) = 1 \text{ for all } x \in X \text{ and}$$
$$u_{d_k}(\pi(d_k)) = 3 \text{ for all } k \in [s-r]. \tag{3.2}$$

Since (3.2) is easy to check, this equivalence implies that $\mathcal{A}$ can be used to solve X3C, and hence our problem is NP-hard. If (3.2) holds, then the argument in the proof of Theorem 3 applies and shows that the X3C instance has a solution.

Conversely, suppose there is a solution $\mathcal{S}' \subseteq \mathcal{S}$ to the X3C instance. Then, as in the proof of Theorem 3, there is an allocation $\pi^* : N_o \to \mathcal{C}(V_o)$ of the original items to the original agents such that $u_{i_x}(\pi^*(i_x)) = 1$ for all $x \in X$ and $u_{d_k}(\pi^*(d_k)) = 3$ for all $k \in [s-r]$. Extend π^* to all agents by defining $\pi^*(\hat{j}) = \pi(\hat{j}) \cap (V \setminus V_o)$ for every copied agent $\hat{j}$. It is easy to check that π^* is a connected allocation. For each copied agent $\hat{j}$, we have $u_{\hat{j}}(\pi^*(\hat{j})) = u_{\hat{j}}(\pi(\hat{j}))$, since $\hat{j}$ has a valuation of 0 for every item in V_o. Also, for each original agent $i \in N_o$, we have $u_i(\pi^*(i)) \geqslant u_i(\pi(i))$, since i obtains an optimal bundle under π^*. It follows that if π fails (3.2), then π^* is a Pareto-improvement of π, contradicting that π is Pareto-optimal. So π satisfies (3.2). $\square$

Note that the graph constructed in the above proof has pathwidth 2 and diameter 4, so hardness holds even for trees of bounded pathwidth and bounded diameter. One can adapt our reduction to show that hardness holds on trees with maximum degree 3, by copying our original reduction many times.

Theorem 5. *Unless* P = NP*, there is no polynomial-time algorithm which finds a Pareto-optimal connected allocation when G is a tree with maximum degree 3, even if valuations are binary and additive.*

In the last section, we saw positive results for paths and stars when valuations are additive. For more general preferences over bundles, we again obtain a hardness result. Refer to the full version of this paper for definitions of 2-additive and of dichotomous valuations.

Theorem 6. *Unless* P = NP*, there is no polynomial-time algorithm which finds a Pareto-optimal connected allocation when G is a path, when valuations are 2-additive. The problem is also hard when G is a star and valuations are 2-additive. Both problems are also hard for dichotomous valuations specified by a formula of propositional logic.*

4 Pareto-Optimality & EF1 on Paths

In Section 3, we were aiming to find *some* Pareto-optimum, and obtained a positive result for the important case where G is a path. Now we aim higher, wanting to find an efficient allocation which is also fair, where by fairness we mean EF1.

When there are no connectivity requirements, it is known that efficiency and fairness are compatible: Caragiannis et al. (2016) showed that an allocation maximizing the *Nash product* of agents' valuations is both Pareto-optimal and EF1. While it is NP-hard to compute the Nash solution, Barman, Murthy, and Vaish (2018) designed a (pseudo-)polynomial-time algorithm which finds an allocation satisfying these two properties.

In our model, unfortunately, EF1 is incompatible with Pareto-optimality, even when G is a path. The following examples only require binary additive valuations and only two or three agents. Note that Bilò et al. (2018) proved that an EF1 allocation always exists on a path for up to four agents. Also, it follows from results of Barrera et al. (2015) that an EF1 allocation always exists on a path with binary additive valuations and any number of agents.

Example 1. Consider an instance with two agents a, b and a path with five items $v_1, \ldots, v_5$, and binary additive valuations as shown below.

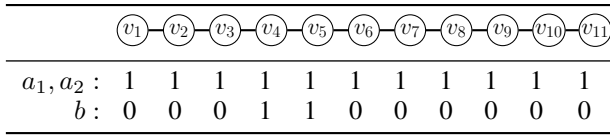|	v_1	v_2	v_3	v_4	v_5
$a:$	1	0	1	1	0
$b:$	0	1	1	0	1

Write an allocation π as a pair $(\pi(a), \pi(b))$, omitting set braces for brevity. Then the allocation

- $(v_1, v_2v_3v_4v_5)$ is not EF1,
- $(v_1v_2, v_3v_4v_5)$ is Pareto-dominated by $(v_1, v_2v_3v_4v_5)$,
- $(v_1v_2v_3, v_4v_5)$ is Pareto-dominated by $(v_1v_2v_3v_4, v_5)$,
- $(v_1v_2v_3v_4, v_5)$ is not EF1.

The other allocations also fail Pareto-optimality or EF1, by symmetry. $\square$

The following alternative example shows that Pareto-optimality and EF1 conflict in another restricted setting, where each agent's approval set is an interval.

Example 2. Consider an instance with three agents $a_1, a_2,$ and b, and a path with eleven items $v_1, \ldots, v_{11}$, and binary additive valuations as shown below.

(v_1)—(v_2)—(v_3)—(v_4)—(v_5)—(v_6)—(v_7)—(v_8)—(v_9)—(v_{10})—(v_{11})

$a_1, a_2:$	1	1	1	1	1	1	1	1	1	1	1
$b:$	0	0	0	1	1	0	0	0	0	0	0

Suppose π is a Pareto-optimal EF1 allocation. Then, for each $i = 1, 2$, because b does not envy a_i up to one good, we have $\{v_4, v_5\} \not\subseteq \pi(a_i)$. Thus, for each $i = 1, 2$, we have either $\pi(a_i) \subseteq \{v_1, \ldots, v_4\}$ (and we say a_i is in group L) or $\pi(a_i) \subseteq \{v_5, \ldots, v_{11}\}$ (and a_i is in group R). Now, a_1 and a_2 are not both in group L, since then there would be a Pareto-improvement by giving the six items $\{v_6, \ldots, v_{11}\}$ to a_1. Also, a_1 and a_2 are not both in group R, since then one of them (say a_1) would receive at most 3 approved items, and there would be a Pareto-improvement by giving items $\{v_1, v_2, v_3\}$ to a_1 and $\{v_6, \ldots, v_{11}\}$ to a_2. Hence, without loss of generality, a_1 is in group L and a_2 is in group R. Since π is Pareto-optimal, we have $\pi(b) \subseteq \{v_4, v_5\}$; if b were to obtain any other items, then we can reallocate these items to a_1 and a_2 to obtain a Pareto-improvement. Thus, a_1 obtains at most four approved items (as $\pi(a_1) \subseteq \{v_1, \ldots, v_4\}$), but a_2 receives at least six approved items (as $\{v_6, \ldots, v_{11}\} \subseteq \pi(a_2)$), so π is not EF1, a contradiction. $\square$

Given that we do not have an existence guarantee, a natural question is whether it is easy to decide whether a given instance admits a Pareto-optimal allocation satisfying EF1. Using the above examples, we prove that the problem is NP-hard. The obvious complexity upper bound is Σ_2^p; an open problem is whether the problem is complete for this class. A related result of de Keijzer et al. (2009) shows that without connectivity constraints and with additive valuations, it is Σ_2^p-complete to decide whether a Pareto optimal and envy-free allocation exists; see also Bouveret and Lang (2008).

Theorem 7. *It is NP-hard to decide whether a connected allocation that is Pareto-optimal and satisfies EF1 exists when G is a path, even if valuations are binary and additive.*

Note that in Examples 1 and 2, there are two different types of agents' valuations. Invoking a recent result independently obtained by Bilò et al. (2018) and Oh, Procaccia, and Suksompong (2019), we can show that a Pareto-optimal EF1 allocation exists on paths for agents with additive valuations that are *identical*, i.e., $u_i(X) = u_i(X)$ for all bundles $X \in \mathcal{C}(V)$ and all $i, j \in N$.

Proposition 1. *When G is a path and agents have identical additive valuations, a connected allocation that is Pareto-optimal and satisfies EF1 exists and can be found efficiently.*

Proof. When agents have identical additive valuations, every allocation π has the same utilitarian social welfare $\sum_{i \in N} u_i(\pi(i)) = \sum_{v \in V} u_1(v)$. Hence, every allocation maximizes social welfare and is thus Pareto-optimal. Now, Bilò et al. (2019, Theorem 7.1) and Oh, Procaccia, and Suksompong (2019, Lemma C.2) show that if G is a path, a connected EF1 allocation exists, which, by the above reasoning, is also Pareto-optimal. This allocation can be found efficiently since the existence results of Bilò et al. (2018) and Oh, Procaccia, and Suksompong (2019) both come with an efficient algorithm. $\square$

For identical valuations that are not additive, Pareto-optimality and EF1 are again incompatible on a path.

Example 3. There are four items a, b, c, d arranged on a path, and two agents with the following identical valuations:

X	$u(X)$	X	$u(X)$	X	$u(X)$
$\{a\}$	2	$\{a, b\}$	2	$\{a, b, c\}$	3
$\{b\}$	2	$\{b, c\}$	3	$\{b, c, d\}$	4
$\{c\}$	2	$\{c, d\}$	3	$\{a, b, c, d\}$	4
$\{d\}$	1				

These valuations are subadditive. Then the allocation

- $\{\{a, b, c, d\}, \emptyset\}$ is not EF1,
- $\{\{a, b, c\}, \{d\}\}$ is not EF1,
- $\{\{a, b\}, \{c, d\}\}$ is Pareto-dominated by $\{\{a\}, \{b, c, d\}\}$,
- $\{\{a\}, \{b, c, d\}\}$ is not EF1. $\square$

5 Pareto-Optimality & MMS on Paths

In the previous section, we saw that deciding the existence of an allocation that is Pareto-efficient and satisfies EF1 is computationally hard, even for a path, and saw examples where no such allocation exists. Part of the reason is that envy-freeness notions and Pareto-optimality are not natural companions: it is easy to construct envy-free allocations, which, after a Pareto-improvement, are not envy-free anymore.

An alternative notion of fairness avoids this problem: Pareto-improving upon an MMS allocation preserves the MMS property, because MMS only specifies a lower bound on agents' utilities. Bouveret et al. (2017) showed that if G is a tree, then an MMS allocation always exists (and can be found efficiently). Hence, if G is a tree, there is an allocation that is both Pareto-optimal and MMS: take an MMS allocation, and repeatedly find Pareto-improvements until reaching a Pareto-optimum, which must still satisfy the MMS property.

While existence is guaranteed, it is unclear whether we can find an allocation satisfying both properties in polynomial time. Certainly, by the negative result of Theorem 4, this is not possible when G is an arbitrary tree. What about the case when G is a path? The answer is also negative: a Pareto-optimal MMS allocation cannot be found efficiently.

Theorem 8. *Unless* P = NP, *there is no polynomial-time algorithm which finds a Pareto-optimal MMS allocation when G is a path, even if valuations are binary and additive.*

Proof. We again give a Turing reduction from X3C, building on the reduction of Theorem 3. Suppose we are given an instance $(X, \mathcal{S})$ of X3C, where $X = \{x_1, x_2, \ldots, x_{3r}\}$ and $\mathcal{S} = \{S_1, \ldots, S_s\}$. Construct the paths $P_{S_1}, P_{S_2}, \ldots, P_{S_s}$ and agents i_x for each $x \in X$ and d_k for each $k \in [s - r]$ with binary utilities like in the proof of Theorem 3. We write $N_o = \{i_x : x \in X\} \cup \{d_1, d_2, \ldots, d_{s-r}\}$ and $V_o = \bigcup_{S \in \mathcal{S}} \{v_{S,1}, v_{S,2}, v_{S,3}\}$ for the sets of agents and items introduced so far. In addition, for each $k \in [s]$, we construct a path B_k of $2r + 2s$ new vertices $b_k^1, b_k^2, \ldots, b_k^{2r+2s}$. The graph G is obtained by concatenating these paths in the order $P_1, B_1, \ldots, P_s, B_s$. Finally, for each $k \in [s]$, we introduce an agent z_k who approves exactly the vertices on B_k. The agents in N_o do not approve any of the items in $B_1, \ldots, B_s$.

Note that, in total, there are $3r + (s - r) + s = 2r + 2s$ agents. Since each agent z_k approves $2r + 2s$ vertices, each agent z_k has positive MMS value, namely $\mathrm{mms}_{z_k} = 1$.

Suppose we had an algorithm $\mathcal{A}$ which finds a Pareto-optimal MMS allocation on a path. We show how to use $\mathcal{A}$ to solve X3C. Run $\mathcal{A}$ on the allocation problem constructed above to obtain a Pareto-optimum π which satisfies MMS. Then, for each $k \in [s]$, the agent z_k receives at least one vertex from B_k since π is MMS. It follows that no agent $i \in N_0$ can receive items from two different paths P_{S_j} and P_{S_k}, $j < k$, since these paths are separated by B_j. Thus, for each $i \in N_0$, there is some $j \in [s]$ with $\pi(i) \subseteq B_{j-1} \cup P_{S_j} \cup B_j$. By suitably reallocating items that agent i does not approve, we can in fact assume that $\pi(i) \subseteq P_{S_j}$ for some $j \in [s]$. This implies that $u_{i_x}(\pi(i_x)) \leqslant 1$ for all $x \in X$ and $u_{d_k}(\pi(d_k)) \leqslant 3$ for all $k \in [s - r]$.

We now prove that the X3C instance has a solution iff

$$
\begin{aligned}
u_{i_x}(\pi(i_x)) &= 1 \text{ for all } x \in X \text{ and} \\
u_{d_k}(\pi(d_k)) &= 3 \text{ for all } k \in [s - r].
\end{aligned}
\tag{5.1}
$$

Since (5.1) is easy to check, this equivalence implies that $\mathcal{A}$ can be used to solve X3C, and hence our problem is NP-hard.

If (5.1) holds, then the argument in the proof of Theorem 3 applies and shows that the X3C instance has a solution.

Conversely, suppose there is a solution $\mathcal{S}' \subseteq \mathcal{S}$ to the X3C instance. Then, as in the proof of Theorem 3, there is an allocation $\pi^* : N_o \to \mathcal{C}(V_o)$ of the original items to the original agents such that $u_{i_x}(\pi^*(i_x)) = 1$ for all $x \in X$ and $u_{d_k}(\pi^*(d_k)) = 3$ for all $k \in [s - r]$. Extend π^* to all agents by defining $\pi^*(z_k) = B_k$ for each $k \in [s]$. It is easy to check that π^* is a connected allocation. For each $k \in [s]$, we have $u_{z_k}(z_k) = u_{z_k}(\pi(z_k))$, since z_k receives all approved items in π^* Also, for each original agent $i \in N_o$, we have $u_i(\pi^*(i)) \geqslant u_i(\pi(i))$, since i obtains an optimal bundle under π^*. Hence, if π fails (5.1), then π^* is a Pareto-improvement of π, a contradiction. So π satisfies (5.1). $\square$

For $\alpha \in (0, 1]$, we say that an allocation π is α-MMS if $u_i(\pi(i)) \geqslant \alpha \cdot \mathrm{mms}_i$ for all $i \in N$. The above proof implies that we cannot in polynomial time find a Pareto-optimal allocation that is α-MMS, for fixed $\alpha > 0$. The reduction can also easily be adapted to the case when G is a cycle.

Next, we show that when G is a path, we can find a Pareto-optimal MMS allocation in polynomial time for a restricted class of valuations. We assume that agents' valuations are binary and additive, and for each voter, the set of approved vertices forms an interval of the path G, and finally these intervals are *non-nested*. Formally, for agent $i \in N$, we let $A(i) = \{v \in V : u_i(v) = 1\}$ be the set of vertices which i approves. For a path $P = (1, 2, \ldots, m)$, we say that binary and additive valuations are *given by non-nested intervals* if $A(i)$ is connected on the path for each $i \in N$, and there is no pair of agents $i, j \in N$ with $\min A(i) < \min A(j)$ and $\max A(j) < \max A(i)$. This restriction is plausible, for instance, when several groups wish to book the same conference venue; each group specifies a period of contiguous dates of (almost) equal length that are suitable for them.
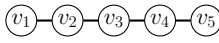
We show that when valuations have this form, there is a polynomial-time algorithm which yields an MMS allocation

that is Pareto-optimal. The algorithm is an adaptation of the moving-knife algorithm of Bouveret et al. (2017). The non-nestedness assumption means that agents can be ordered naturally from left to right. Our algorithm then allocates bundles to agents in that order. In the process, we can ensure that every item has been allocated to an agent who approves of it. Hence, the resulting allocation maximizes utilitarian social welfare and thus is Pareto-optimal.

Theorem 9. *When G is a path and valuations are binary and additive given by non-nested intervals, there exists a polynomial-time algorithm that finds a connected MMS allocation that maximizes utilitarian social welfare.*

As the following example shows, the non-nestedness assumption is necessary for the result of Theorem 9 to hold.

Example 4. Consider an instance with two agents and five vertices on a path, with binary additive valuations as below.

(v_1)—(v_2)—(v_3)—(v_4)—(v_5)				
Alice: 1	1	1	1	1
Bob: 0	1	1	0	0

The unique connected allocation maximizing utilitarian welfare is the allocation giving all items to Alice, which violates the MMS requirement for Bob. $\square$

6 Conclusion

In this work, we have studied the computational complexity of finding Pareto-efficient outcomes, in the natural setting where we need to allocate indivisible items into connected bundles. We showed that although finding a Pareto-optimal allocation is easy for some topologies, this does not extend to general trees. Further, we proved that when imposing additional fairness requirements, finding a Pareto-optimum becomes NP-hard even when the underlying item graph is a path. We have also seen that a Pareto-optimal EF1 allocation may not exist with the contiguity requirement while such an allocation always exists when these requirements are ignored.

While we have focused on the divisions of goods, studying an allocation of chores (bads) with graph-connectivity constraints is an interesting future direction. In particular, one may ask what graph structures give positive results in terms of both existence and computational complexity. Some of the questions studied in this paper are of interest also in the setting without connectivity constraints: What is the complexity of finding a Pareto-optimum with non-additive valuations? Are there polynomial-time algorithms finding a Pareto-optimal α-MMS allocation for constant $\alpha > 0$?

References

Aumann, Y., and Dombb, Y. 2015. The efficiency of fair division with connected pieces. *ACM Transactions on Economics and Computation* 3:23:1–23:16.

Aumann, Y.; Dombb, Y.; and Hassidim, A. 2013. Computing socially-efficient cake divisions. In *Proceedings of the 12th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, 343–350.

Aziz, H.; Bíró, P.; Lang, J.; Lesca, J.; and Monnot, J. 2016. Optimal reallocation under additive and ordinal preferences. In *Proceedings of the 15th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, 402–410.

Aziz, H.; Brandt, F.; and Harrenstein, P. 2013. Pareto optimality in coalition formation. *Games and Economic Behavior* 82:562–581.

Barman, S.; Murthy, S. K. K.; and Vaish, R. 2018. Finding fair and efficient allocations. In *Proceedings of the 19th ACM Conference on Economics and Computation (ACM-EC)*, 557–574.

Barrera, R.; Nyman, K.; Ruiz, A.; Su, F. E.; and Zhang, Y. X. 2015. Discrete envy-free division of necklaces and maps. *arXiv:1510.02132 [math.CO]*.

Bei, X.; Chen, N.; Hua, X.; Tao, B.; and Yang, E. 2012. Optimal proportional cake cutting with connected pieces. In *Proceedings of the 26th AAAI Conference on Artificial Intelligence (AAAI)*, 1263–1269.

Bilò, V.; Caragiannis, I.; Flammini, M.; Igarashi, A.; Monaco, G.; Peters, D.; Vinci, C.; and Zwicker, W. S. 2018. Almost envy-free allocations with connected bundles. In *Proceedings of the 10th Innovations in Theoretical Computer Science Conference (ITCS 2019)*, 14:1–14:21. Full version arXiv:1808.09406 [cs.GT].

Bouveret, S., and Lang, J. 2008. Efficiency and envy-freeness in fair division of indivisible goods: Logical representation and complexity. *Journal of Artificial Intelligence Research* 32:525–564.

Bouveret, S.; Cechlárová, K.; Elkind, E.; Igarashi, A.; and Peters, D. 2017. Fair division of a graph. In *Proceedings of the 24th International Joint Conference on Artificial Intelligence (IJCAI)*, 135–141.

Budish, E. 2011. The combinatorial assignment problem: Approximate competitive equilibrium from equal incomes. *Journal of Political Economy* 119(6):1061–1103.

Caragiannis, I.; Kurokawa, D.; Moulin, H.; Procaccia, A. D.; Shah, N.; and Wang, J. 2016. The unreasonable fairness of maximum Nash welfare. In *Proceedings of the 17th ACM Conference on Economics and Computation (ACM-EC)*, 305–322.

Conitzer, V.; Derryberry, J.; and Sandholm, T. 2004. Combinatorial auctions with structured item graphs. In *Proceedings of the 19th AAAI Conference on Artificial Intelligence (AAAI)*, 212–218.

de Keijzer, B.; Bouveret, S.; Klos, T.; and Zhang, Y. 2009. On the complexity of efficiency and envy-freeness in fair division of indivisible goods with additive preferences. In *Proceedings of the 1st International Conference on Algorithmic Decision Theory (ADT)*, 98–110. Springer.

Igarashi, A., and Peters, D. 2018. Pareto-optimal allocation of indivisible goods with connectivity constraints. *arXiv:1811.04872 [cs.GT]*.

Korte, B., and Vygen, J. 2006. *Combinatorial Optimization - Theory and Algorithms*. Springer.

Lindner, C., and Rothe, J. 2016. Cake-cutting: Fair division of divisible goods. In *Economics and Computation*. Springer. 395–491.

Lonc, Z., and Truszczynski, M. 2018. Maximin share allocations on cycles. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence (IJCAI)*, 410–416.

Oh, H.; Procaccia, A.; and Suksompong, W. 2019. Fairly allocating many goods with few queries. In *Proceedings of the 33rd AAAI Conference on Artificial Intelligence (AAAI)*. Full version arXiv:1807.11367v1 [cs.GT].

Stromquist, W. 1980. How to cut a cake fairly. *American Mathematical Monthly* 87:640–644.

Stromquist, W. 2008. Envy-free divisions cannot be found by finite protocols. *The Electronic Journal of Combinatorics* 15:145–158.

Suksompong, W. 2017. Fairly allocating contiguous blocks of indivisible items. In *Proceedings of the 10th International Symposium on Algorithmic Game Theory (SAGT)*, 333–344.