Fair allocation of combinations of indivisible goods and chores

Haris Aziz haris.aziz@unsw.edu.au
UNSW Sydney and Data61 CSIRO, Australia
Ioannis Caragiannis caragian@ceid.upatras.gr
University of Patras, Greece
Ayumi Igarashi ayumi_igarashi@nii.ac.jp
National Institute of Informatics, Japan
Toby Walsh tw@cse.unsw.edu.au
UNSW Sydney and Data61 CSIRO, Australia

Abstract

We consider the problem of fairly dividing a set of items. Much of the fair division literature assumes that the items are "goods" i.e., they yield positive utility for the agents. There is also some work where the items are "chores" that yield negative utility for the agents. In this paper, we consider a more general scenario where an agent may have negative or positive utility for each item. This framework captures, e.g., fair task assignment, where agents can have both positive and negative utilities for each task. We show that whereas some of the positive axiomatic and computational results extend to this more general setting, others do not. We present several new and efficient algorithms for finding fair allocations in this general setting. We also point out several gaps in the literature regarding the existence of allocations satisfying certain fairness and efficiency properties and further study the complexity of computing such allocations. [1]

1. Introduction

Consider a group of students who are assigned to a certain set of coursework tasks. Students may have subjective views regarding how enjoyable each task is. For some people, solving a mathematical problem may be fulfilling and rewarding. For others, it may be nothing but torture. A student who gets more cumbersome chores may be compensated by giving her some valued goods so that she does not feel hard done by.

This example can be viewed as an instance of a classic fair division problem. The agents have different preferences over the items and we want to allocate the items to agents as fair as possible. The twist we consider is that whether an agent has positive or negative utility for an item is subjective. Our setting is general enough to encapsulate two well-studied settings: (1) "good allocation" in which agents have positive utilities for the items and (2) "chore allocation" in which agents have negative utilities for the items. The setting we

1. This article is a complete version of a conference paper, which appeared in the Proceedings of the 28th International Joint Conference on Artificial Intelligence (IJCAI) (Aziz, Caragiannis, Igarashi, & Walsh, 2019). The conference version contains an error in the proof of Theorem 2 regarding EF1 existence for arbitrary utility functions. In this extended version, we fix this error by weakening the theorem statement to the class of doubly monotonic utilities, and provide all proofs that were omitted from the conference version.

consider also covers a third setting (3) "allocation of objective goods and chores" in which the items can be partitioned into chores (that yield negative utility for all agents) and goods (that yield positive utility for all agents). Setting (3) covers several scenarios where an agent could be compensated by some goods for doing some chores.

In this paper, we suggest a very simple yet general model of allocation of indivisible items that properly includes chore and good allocation. For this model, we present some case studies that highlight that whereas some existence and computational results can be extended to our general model, in other cases the combination of good and chore allocation poses interesting challenges not faced in subsettings. Our central technical contributions are several new efficient algorithms for finding fair allocations. In particular:

- We formalize fairness concepts for the general setting. Some fairness concepts directly extend from the setting of good allocation to our setting. Other fairness concepts such as "envy-freeness up to one item" (EF1) and "proportionality up to one item" (PROP1) need to be generalized appropriately.

- We show a careful generalization of the decentralized round robin algorithm that finds an EF1 allocation when utilities are additive.

- We present a different polynomial-time algorithm that always returns an EF1 allocation even when the agents' utility functions are doubly monotonic (but not necessarily additive).

- Turning our attention to an efficient and fair allocation, we show that for the case of two agents, there exists a polynomial-time algorithm that finds an EF1 and Pareto-optimal (PO) allocation for our setting. The algorithm can be viewed as an interesting generalization of the Adjusted Winner rule (Brams & Taylor, 1996a, 1996b) that is designed for divisible goods.

- If we weaken EF1 to PROP1, then we show that there exists an allocation that is not only PROP1 but is also contiguous (assuming that items are placed in a line). We further give a polynomial-time algorithm that finds such an allocation.

1.1 Related Work

Fair allocation of indivisible items is a central problem in several fields including computer science and economics (Aziz, Gaspers, Mackenzie, & Walsh, 2015; Brams & Taylor, 1996a; Bouveret, Chevaleyre, & Maudet, 2016; Lipton, Markakis, Mossel, & Saberi, 2004). Fair allocation has been extensively studied for allocation of divisible goods, commonly known as cake cutting (Brams & Taylor, 1996a).

There are several established notions of fairness, including envy-freeness and proportionality. The recently introduced *maximin share* (MMS) notion is weaker than envy-freeness and proportionality and has been heavily studied in the computer science literature. Kurokawa et al. (2018) showed that an MMS allocation of goods may not always exist; on the positive side, there exists a polynomial-time algorithm that returns a 2/3-approximate MMS allocation (Kurokawa et al., 2018; Amanatidis, Markakis, Nikzad, & Saberi, 2017). Subsequent papers have presented simpler (Barman & Krishnamurthy, 2020) or even better

(Ghodsi, HajiAghayi, Seddighin, Seddighin, & Yami, 2018) approximation algorithms for MMS allocations. In general, checking whether there exists an envy-free and Pareto-optimal allocation for goods is Σ_2^p-complete (de Keijzer, Bouveret, Klos, & Zhang, 2009).

The idea of envy-freeness up to one good (EF1) was implicit in the paper by Lipton et al. (2004). Today, it has become a well-studied fairness concept in its own right (Budish, 2011). Caragiannis et al. (2019) further popularized it, showing that a natural modification of the Nash welfare maximizing rule satisfies EF1 and PO for the case of goods. Barman et al. (2018) recently presented a pseudo-polynomial-time algorithm for computing an allocation that is PO and EF1 for goods. A stronger fairness concept, *envy freeness up to the least valued good* (EFX), was introduced by Caragiannis et al. (2019).

Aziz (2016) noted that the work on multi-agent chore allocation is less developed than that of goods and that results from one may not necessarily carry over to the other. Aziz et al. (2017) considered fair allocation of indivisible chores and showed that there exists a simple polynomial-time algorithm that returns a 2-approximate MMS allocation for chores. Barman and Krishnamurthy (2020) presented a better approximation algorithm. Caragiannis et al. (2012) studied the efficiency loss in order to achieve several fair allocations in the context of both good and chore divisions. Allocation of a mixture of goods and chores has received recent attention in the context for divisible items (Bogomolnaia, Moulin, Sandomirskyi, & Yanovskaya, 2016, 2017). Here, we focus on indivisible items.

2. Our Model and Fairness Concepts

We now define a fair division problem of indivisible items where agents may have both positive and negative utilities. For a natural number $s \in \mathbb{N}$, we write $[s] = \{1, 2, \ldots, s\}$. An *instance* is a triple $I = (N, O, U)$ where

- $N = [n]$ is a set of *agents*,
- $O = \{o_1, o_2, \ldots, o_m\}$ is a set of *indivisible items*, and
- U is an n-tuple of utility functions $u_i : O \to \mathbb{R}$.

We note that under this model, an item can be a good for one agent (i.e., $u_i(o) > 0$) but a chore for another agent (i.e., $u_j(o) < 0$). For $X \subseteq O$, we write $u_i(X) := \sum_{o \in X} u_i(o)$; we assume that the utilities in this paper are additive unless specified otherwise. Each subset $X \subseteq O$ is referred to as a *bundle* of items. An *allocation* π is a function $\pi : N \to 2^O$ assigning each agent a different bundle of items, i.e., for every pair of distinct agents $i, j \in N$, $\pi(i) \cap \pi(j) = \emptyset$; it is said to be *complete* if $\bigcup_{i \in N} \pi(i) = O$.

We first observe that the definitions of standard fairness concepts can be naturally extended to this general model. The most classical fairness principle is *envy-freeness*, requiring that agents do not envy each other. Specifically, given an allocation π, we say that i *envies* j if $u_i(\pi(i)) < u_i(\pi(j))$. An allocation π is *envy-free* (EF) if no agent envies the other agents. Another appealing notion of fairness is *proportionality* which guarantees each agent an $1/n$ fraction of her utility for the whole set of items. Formally, an allocation π is *proportional* (PROP) if each agent $i \in N$ receives a bundle $\pi(i)$ of value at least her *proportional fair share* $u_i(O)/n$. The following implication, which is well-known for the case of goods, holds in our setting as well.

Proposition 1. *For additive utilities, an envy-free complete allocation satisfies proportionality.*

Proof. Suppose that an allocation π is an envy-free allocation. Then for each $i \in N$, $u_i(\pi(i)) \geq u_i(\pi(j))$ for all $j \in N$. Thus, by summing up all the inequalities, $n \cdot u_i(\pi(i)) \geq \sum_{j \in N} u_i(\pi(j)) = u_i(O)$. Hence each $i \in N$ receives a bundle of value at least $u_i(O)/n$, so π satisfies proportionality. □

A simple example of one good with two agents already suggests the impossibility in achieving envy-freeness and proportionality. The recent literature on indivisible allocation has, thereby, focused on approximations of these fairness concepts. A prominent relaxation of envy-freeness, introduced by Budish (2011), is *envy-freeness up to one good* (EF1), which requires that an agent's envy towards another bundle can be eliminated by removing some good from the envied bundle. We will present a generalized definition for EF1 that has only been considered in the context of good allocation: the envy can diminish by removing either one "good" from the other's bundle or one "chore" from their own bundle. Given an allocation π, we say that i *envies* j up to one item if i does not envy j, or there is an item $o \in \pi(i) \cup \pi(j)$ such that $u_i(\pi(i) \setminus \{o\}) \geq u_i(\pi(j) \setminus \{o\})$.

Definition 2 (EF1)**.** An allocation π is *envy-free up to one item (EF1)* if for all $i, j \in N$, i envies j up to one item.

Obviously, envy-freeness implies EF1. Conitzer et al. (2017) introduced a novel relaxation of proportionality, which they called *PROP1*. In the context of good allocation, this fairness relaxation is a weakening of both EF1 and proportionality, requiring that each agent gets her proportional fair share if she obtains one additional good from the others' bundles. Now we will extend this definition to our setting: under our definition, each agent receives her proportional fair share by obtaining an additional good or removing some chore from her bundle.

Definition 3 (PROP1)**.** An allocation π satisfies *proportionality up to one item (PROP1)* if for each agent $i \in N$,

- $u_i(\pi(i)) \geq u_i(O)/n$; or
- $u_i(\pi(i)) + u_i(o) \geq u_i(O)/n$ for some $o \in O \setminus \pi(i)$; or
- $u_i(\pi(i)) - u_i(o) \geq u_i(O)/n$ for some $o \in \pi(i)$.

It can be easily verified that EF1 implies PROP1.

Proposition 4. *For additive utilities, an EF1 complete allocation satisfies PROP1.*

Proof. Suppose π satisfies EF1. Consider any agent $i \in N$. Let $x = \max_{o \in O \setminus \pi(i)} u_i(o)$ and $y = -\min_{o \in \pi(i)} u_i(o)$. Since π satisfies EF1, if i gets bonus value b_i by removing getting some good or some chore where $b_i = \max\{x, y, 0\}$, her updated utility is such that $u_i(\pi(i)) + b_i \geq u_i(\pi(j))$ for any agent j. This would imply that $n(u_i(\pi(i)) + b_i) \geq \sum_{j \in N} u_i(\pi(j)) = u_i(O)$, which implies that $u_i(\pi(i)) + b_i \geq u_i(O)/n$. Hence PROP1 is satisfied. □

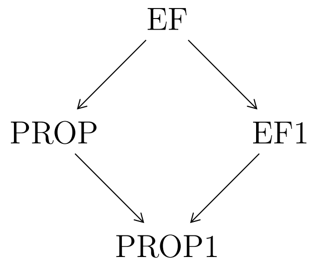

Figure 1: Relations between fairness concepts

Figure 1 illustrates the relations between fairness concepts introduced above.

Besides fairness, we will also consider an efficiency criterion. The most commonly used efficiency concept is *Pareto-optimality*. Given an allocation π, another allocation π' is a *Pareto-improvement* of π if $u_i(\pi'(i)) \geq u_i(\pi(i))$ for all $i \in N$ and $u_j(\pi'(j)) > u_j(\pi(j))$ for some $j \in N$. We say that an allocation π is *Pareto-optimal* (PO) if there is no allocation that is a Pareto-improvement of π.

3. Finding an EF1 Allocation

Double Round Robin Algorithm In this section, we focus on EF1, a very permissive fairness concept that admits a polynomial-time algorithm in the case of good allocation. For instance, consider a *round robin rule* in which agents take turns, and choose their most preferred unallocated item. The round robin rule finds an EF1 allocation if all the items are goods (see e.g., Caragiannis et al. (2019)). By a very similar argument, it can be shown that the algorithm also finds an EF1 allocation if all the items are chores. However, we will show that the round robin rule already fails to find an EF1 allocation if we have some items that are goods and others that are chores.

Proposition 5. *The round robin rule does not satisfy EF1.*

Proof. Suppose there are two agents and four items with identical utilities described below.

	①	②	③	④
Alice, Bob:	2	-3	-3	-3

Consider the order, in which Alice chooses the only good and then the remaining chores of equal value are allocated accordingly. In that case, Alice gets the positive-value and one chore, whereas Bob gets two chores. So even if one item is removed from the bundles of Alice or Bob, Bob will still remain envious. □

Nevertheless, a careful adaptation of the round robin method to our setting, which we call the *double round robin algorithm*, constructs an EF1 allocation. In essence, the algorithm will apply the round robin method twice: clockwise and anticlockwise. In the first phase, the round-robin algorithm allocates *chores* to agents (i.e., the items for which each agent has non-positive utility), while in the second phase, the reversed round-robin algorithm allocates the remaining *goods* to agents , in the opposite order starting with the agent who is worst off in the first phase. Intuitively each agent i may envy agent j who comes earlier than her at the end of one phase, but i does not envy j with respect to the items allocated in the other round; hence the envy of i towards j can be bounded up to one item. We formalize the idea in Algorithm 1; see Figure 2 for an illustration.

Algorithm 1 Double Round Robin Algorithm

Input: An instance $I = (N, O, U)$

Output: An allocation π

1: Partition O into $O^+ = \{o \in O \mid \exists i \in N \text{ s.t. } u_i(o) > 0\}$, $O^- = \{o \in O \mid \forall i \in N, u_i(o) \leq 0\}$ and suppose $|O^-| = an - k$ for some integer a and $k \in \{0, , \ldots, n-1\}$.

2: Create k dummy chores for which each agent has utility 0, and add them to O^-. (Hence $|O^-| = an$.)

3: Let the agents come in a robin robin sequence $(1, 2, \ldots, n)^*$ and pick their most preferred item in O^- until all items in O^- are allocated.

4: Let the agents come in a robin robin sequence $(n, n-1, \ldots, 1)^*$ and pick their most preferred item in O^+ until all items in O^+ are allocated. If an agent has no available item which gives her strictly positive utility, she does not get a real item but pretends to *pick* a dummy one for which she has utility 0.

5: Remove the dummy items from the current allocation π and return the resulting allocation π^*.

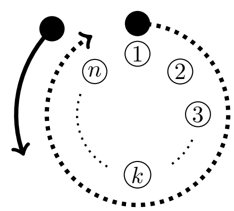

Figure 2: Illustration of Double Round Robin Algorithm. The dotted line corresponds to the picking order when allocating chores. The thick line corresponds to the picking order when allocating goods. The solid black circle indicates the agent who starts with the picking. For the dotted (chores) round, agent 1 is the first agent to pick. For the solid (goods) round, agent n is the first agent to pick.

In the following, for an allocation π and a bundle X, we say that i *envies* j *with respect to* X if $u_i(\pi(i) \cap X) < u_i(\pi(j) \cap X)$.

Theorem 6. *For additive utilities, the double round robin algorithm returns an EF1 complete allocation in* $O(\max\{m^2 \log m, mn\})$ *time.*

Proof. We note that the algorithm ensures that all agents receive the same number of chores, by introducing k virtual chores. Now let π be the output of Algorithm 1. To see that π satisfies EF1, consider any pair of two agents i and j where $i < j$. We will show that by removing one item, these agents do not envy each other.

First, consider i's envy for j. We first observe that the k-th item in O^- allocated to i is weakly preferred by i to the k-th item in O^- allocated to j. Hence, agent i does not envy j with respect to O^-. As for the good allocation, agent i may envy agent j with respect to O^+, which implies that j picks at least one more item from O^+. But if the first item o^* picked by j from O^+ is removed from j's bundle, then the envy will diminish, i.e., i does not envy j with respect to $O^+ \setminus \{o^*\}$. The reason is that for each item in $O^+ \setminus \{o^*\}$ picked by j there is a corresponding item picked by i before j's turn that is at least as preferred by i. Thus $u_i(\pi(i) \geq u_i(\pi(j) \setminus \{o^*\})$.

Second, consider j's envy for i. Similarly to the above, agent j does not envy agent i with respect to O^+ because she takes the first pick among i and j; that is, for every item in $o \in O^+ \cap \pi(i)$ such that $u_j(o) > 0$, agent j picks a corresponding item before i that she weakly prefers. As for the items in O^-, let o^* be the last item from O^- chosen by j. Then, for each item $o \in O^- \setminus \{o^*\}$ picked by i, there is a corresponding item picked by j before i that j weakly prefers to o, which implies that j does not envy i with respect to $O^- \setminus \{o^*\}$. Thus $u_j(\pi(j) \setminus \{o^*\}) \geq u_j(\pi(i))$.

In either case, agents do not envy each other up to one item. We conclude that π is EF1 and so does the final allocation π^* as removing dummy items does not affect the utilities of each agent. It remains to analyze the running time of Algorithm 1. Line 1 requires $O(mn)$ time as each item needs to be examined by all agents. Lines 3 and 4 require $O(m^2 \log m)$ time as there are at most m iterations, and for each iteration, each agent has to choose the most preferred item out of at most m items. Thus, the total running time can be bounded by $O(\max\{m^2 \log m, mn\})$, which completes the proof. □

Generalized Envy Graph Algorithm Algorithm 1 is designed for additive utilities. We construct another algorithm (Algorithm 2) that finds an EF1 allocation for the more general class of *doubly monotonic* utilities. In these utility functions, each agent partitions the items into desirable and undesirable. The value for bundles of items can be arbitrary but with the restriction that adding an item which is desirable for an agent does not decrease her utility and adding an item which is undesirable for an agent does not increase her utility. Formally speaking, agent i's utility function $u_i : 2^O \to \mathbb{R}$ is said to be *doubly monotonic* if agent i can partition the items into two disjoint sets G_i and C_i such that for any item $o \in O$ and for any bundle of items $X \subseteq O \setminus \{o\}$,

- $u_i(X \cup \{o\}) \geq u_i(X)$ if $o \in G_i$; and
- $u_i(X \cup \{o\}) \leq u_i(X)$ if $o \in C_i$.

The algorithm is based on a generalization of an algorithm presented by Lipton et al. (2004) for finding an EF1 allocation for goods. For an allocation π, the *envy-graph* $G(\pi)$ is a directed graph where the vertices is given by the set of agents N, and for any pair of distinct agents $i, j \in N$, there is an arc from i to j if and only if i envies j. For each directed cycle $C = \{i_1, i_2, \ldots, i_k\}$ of the envy graph $G(\pi)$ where i_j envies i_{j+1} for each $j \in [k]$ and $i_{k+1} = i_1$, we may implement an exchange over the cycle, and define the resulting allocation π_C as follows:

$$\pi_C(i) = \begin{cases} \pi(i) \text{ if } i \notin C, \\ \pi(i_{j+1}) \text{ if } i = i_j \in C. \end{cases}$$

Given an allocation π, the *marginal utility* of an item $o \in O \setminus \pi(i)$ to an agent $i \in N$ is defined as $\delta_i(\pi, o) := u_i(\pi(i) \cup \{o\}) - u_i(\pi(i))$.

Algorithm 2 Generalized Envy Graph Algorithm

Input: An instance $I = (N, O, U)$ where each u_i ($i \in N$) is doubly monotonic with partition (G_i, C_i) where for any item $o \in O$ and for any bundle of items $X \subseteq O \setminus \{o\}$, $u_i(X \cup \{o\}) \geq u_i(X)$ if $o \in G_i$ and $u_i(X \cup \{o\}) \leq u_i(X)$ if $o \in C_i$.

Output: An allocation π

1: Initialize allocation $\pi(i) = \emptyset$ for all $i \in N$
2: **for** $o \in O$ **do**
3:    Set $N^+ = \{\, i \in N \mid \delta_i(\pi, o) \geq 0 \,\}$
4:    **if** $N^+ \neq \emptyset$ **then**
5:       Choose an agent $i^* \in N^+$ with no incoming edge in the graph $G(\pi)$ induced by N^+
6:    **else**
7:       Choose an agent $i^* \in N$ with no outgoing edge in $G(\pi)$
8:    **end if**
9:    Update $\pi(i^*) \leftarrow \pi(i^*) \cup \{o\}$
10:    **while** $G(\pi)$ contains a directed cycle C **do**
11:       Update $\pi \leftarrow \pi_C$
12:    **end while**
13: **end for**

Theorem 7. *For doubly monotone utilities, the generalized envy-graph algorithm finds an EF1 complete allocation in $O(mn^3)$ time.*

Proof. By induction, we will prove the following statements:

(i) each time a new item is allocated in lines 5 and 7, the allocation π is EF1; and

(ii) each time the **while** loop in lines 10 – 12 is executed, the allocation π is EF1 and the envy-graph $G(\pi)$ is acyclic.

The base case clearly holds since the initial allocation corresponds to the empty allocation. We assume that the claims hold up until $k-1$ items have been allocated; we will prove that they still hold after we allocate the k-th item o. To see this, let π' denote the allocation that has been computed just after allocating $k-1$ items, and let π denote the allocation that has been computed just after allocating the k-th item o.

To show (i), it suffices to prove that the envy between agent i^* who receives a new item and any other agent is bounded up to one item.

First, suppose that $N^+ \neq \emptyset$. Let G^+ be the envy graph induced by N^+. By the induction hypothesis, the envy-graph is acyclic, which means that G^+ is acyclic as well. Hence, at least one agent i^* has no incoming arc in G^+ and the algorithm can give item o to agent i^*. Agent i^* receives an item that does not decrease her utility so she envies the other agents up to one item by the induction hypothesis. Take any agent $j \in N \setminus \{i^*\}$. We will show that agent j envies i^* up to one item. Consider the following two cases:

- Suppose that j did not envy i^* before allocating o to i^*. Then, it is clear that $u_j(\pi(j)) \geq u_j(\pi'(i)) = u_j(\pi(i) \setminus \{o\})$.

- Suppose that j envied i^* before allocating o to i^*. This means that $\pi'(j) \neq \emptyset$ or $\pi'(i^*) \neq \emptyset$. Also, j does not belong to N^+ since i^* has no incoming arc in G^+, which implies that $\delta_i(\pi', o) < 0$ and hence $o \in C_j$. If j's envy towards i^* at π' disappears by removing an item from the bundle of j, namely, $\min_{o' \in \pi'(j)} u_j(\pi'(j) \setminus \{o'\}) \geq u_j(\pi'(i^*))$, then

$$\begin{aligned}\min_{o' \in \pi(j)} u_j(\pi(j) \setminus \{o'\}) &= \min_{o' \in \pi'(j)} u_j(\pi'(j) \setminus \{o'\}),\\ &\geq u_j(\pi'(i^*)),\\ &\geq u_j(\pi'(i^*) \cup \{o\}) = u_j(\pi(i^*)),\end{aligned}$$

where the second inequality holds due to double monotonicity. If j's envy towards i^* at π' disappears by removing an item from the bundle of i^*, namely, $u_j(\pi'(j)) \geq \min_{o' \in \pi'(i^*)} u_j(\pi'(i^*) \setminus \{o'\})$, then

$$\begin{aligned}u_j(\pi(j)) = u_j(\pi'(j)) &\geq \min_{o' \in \pi'(i^*)} u_j(\pi'(i^*) \setminus \{o'\}),\\ &\geq \min_{o' \in \pi'(i^*)} u_j((\pi'(i^*) \setminus \{o'\}) \cup \{o\}),\\ &\geq \min_{o' \in \pi(i^*)} u_j(\pi(i^*) \setminus \{o'\}),\end{aligned}$$

where the second inequality holds again due to double monotonicity.

Second, suppose that $N^+ = \emptyset$. Then we know that all the agents have negative marginal utility for o and hence consider item o to be undesirable, i.e., $o \in C_i$ for each $i \in N$. Since, by the induction hypothesis, the envy graph $G(\pi)$ is acyclic, at least one agent i^* has no outgoing arcs, which means that we can give item o to agent i^*. Since i^* envied no one before, her envy towards others can be removed by disposing o. Further, each agent $j \in N \setminus \{i^*\}$ envies i^* up to one item, because adding item o to $\pi(i^*)$ does not increase the utility that agent j would derive from the bundle of i^*. We conclude that the allocation π satisfies EF1 just after the algorithm allocates a new item o to agent i^*, which proves (i).

To show (ii), we now focus on the while loop in the algorithm whereby envy cycles are removed by exchanging allocations along the cycle. After exchanging bundles over cycle C, we observe that the agents in the cycle improve their utility, and agents who are not in C envy each agent in C up to one item, as the set of bundles does not change. Hence, the resulting allocation π_C still satisfies EF1. Further, by removing one cycle, we note that each agent in the cycle has her out-degree decreased by one. Furthermore, no agent outside the cycle has her out-degree changed. Hence, after at most n^2 cycles being removed, the envy-graph has no more envy-cycle. This proves (ii).

Similarly to the analysis of Lipton et al. (2004), the running time of the algorithm can be bounded by $O(mn^3)$. To see this, consider each iteration of the **for** loop in lines 2 – 13. In order to find agent i^* who can be assigned to a new item, we consider at most n agents, and for each agent, we check at most n envy-relations. Further, the **while** loop in lines 10 – 12 requires at most n^3 operations, since there are at most n^2 cycles to be removed and the length of the cycle is at most n. Summing over all m iterations, we conclude that the number of operations of the algorithm is bounded by $O(mn^3)$. □

4. Finding an EF1 and PO allocation

We move on to the the next question as to whether fairness is achievable together with efficiency. In the context of good allocation where agents have non-negative additive utilities, Caragiannis et al. (2019) proved that an outcome that maximizes the *Nash welfare* (i.e., the product of utilities) satisfies EF1 and Pareto-optimality simultaneously. The question regarding whether a Pareto-optimal and EF1 allocation exists for chores is unresolved. Starting from an EF1 allocation and finding Pareto improvements, one runs into two challenges: first, Pareto improvements may not necessarily preserve EF1; second, finding Pareto improvements is NP-hard (Aziz, Biro, Lang, Lesca, & Monnot., 2016; de Keijzer et al., 2009). Even if we ignore the second challenge, the question regarding the existence of a Pareto-optimal and EF1 allocation for chores is open.

Next we show that the problem of finding an EF1 and Pareto-optimal allocation is completely resolved for the restricted but important case of two agents. Our algorithm for the problem can be viewed as a discrete version of the well-known Adjusted Winner (AW) rule (Brams & Taylor, 1996a, 1996b). Just like the Adjusted Winner rule, our algorithm finds a Pareto-optimal and EF1 allocation. In contrast to AW that is designed for goods, our algorithm can handle both goods and chores.

Theorem 8. *For two agents with additive utilities, a Pareto-optimal and EF1 complete allocation always exists and can be computed in $O(m^2)$ time.*

Proof. The algorithm begins by giving each subjective item to the agent who considers it as a good; that is, for each item $o \in O$, it allocates o to agent i if $u_i(o) \geq 0$ and $u_j(o) < 0$ where $j \in N \setminus \{i\}$. So, in the following, let us assume that there is no item for which each agent has utility 0. Also, we assume that we have objective items only, i.e., for each item $o \in O$, either o is a good ($u_i(o) > 0$ for each $i \in N$); or o is a chore ($u_i(o) < 0$ for each $i \in N$). Now we call one of the two agents *winner* (denoted by w) and another *loser* (denoted by ℓ).

1. Initially, all goods are allocated to the winner and all chores to the loser.
2. We sort the items in terms of $|u_\ell(o)|/|u_w(o)|$ (monotone non-increasing order), and consider reallocation of the items according to the ordering (from the left-most to the right-most item).
3. When considering a good, we move it from the winner to the loser. When considering a chore, we move it from the loser to the winner. We stop when we find an EF1 allocation from the point of view of the loser. Note that the loser is envious up to one item of the winner.

We will first prove that at any point of the algorithm, the allocation π is Pareto-optimal, and so is the final allocation π^*. Assume towards a contradiction that the allocation π is Pareto-dominated by the allocation π'. For each $i, j \in \{w, \ell\}$ with $i \neq j$, let

- G_{ii} be the set of goods in $\pi(i) \cap \pi'(i)$;
- C_{ii} be the set of chores in $\pi(i) \cap \pi'(i)$;
- G_{ij} be the set of goods in $\pi(i) \cap \pi'(j)$;

- C_{ij} be the set of chores in $\pi(i) \cap \pi'(j)$.

Without loss of generality, we assume that, in π, the winner has utility which is at least as high as in π', while the loser is strictly better off. Taking into account that the bundles of goods G_{ww} and $G_{\ell\ell}$ and the bundles of chores C_{ww} and $C_{\ell\ell}$ are allocated to the same agent in both allocations, this means

$$u_w(G_{\ell w}) + u_w(C_{\ell w}) - u_w(G_{w\ell}) - u_w(C_{w\ell}) \geq 0; \text{ and} \tag{1}$$
$$u_\ell(G_{w\ell}) + u_\ell(C_{w\ell}) - u_\ell(G_{\ell w}) - u_\ell(C_{\ell w}) > 0 \tag{2}$$

The crucial observation now is that the algorithm considered all items in $G_{\ell w}$ and $C_{w\ell}$ before the items in $G_{w\ell}$ and $C_{\ell w}$ in the ordering (this is why the allocation of the items in the first two bundles changes while the allocation of the items in the last two bundles does not). Let α be such that

$$\max_{o \in G_{w\ell} \cup C_{\ell w}} \frac{|u_\ell(o)|}{|u_w(o)|} \leq \alpha \leq \min_{o \in G_{\ell w} \cup C_{w\ell}} \frac{|u_\ell(o)|}{|u_w(o)|}.$$

This definition implies the inequalities,

$$u_\ell(G_{w\ell}) \leq \alpha u_w(G_{w\ell}); u_\ell(G_{\ell w}) \geq \alpha u_w(G_{\ell w});$$
$$-u_\ell(C_{w\ell}) \geq -\alpha u_w(C_{w\ell}); -u_\ell(C_{\ell w}) \leq -\alpha u_w(C_{\ell w}),$$

which, together with inequality (2), yields

$$\begin{aligned} 0 &< u_\ell(G_{w\ell}) + u_\ell(C_{w\ell}) - u_\ell(G_{\ell w}) - u_\ell(C_{\ell w}) \\ &\leq -\alpha(u_w(G_{\ell w}) + u_w(C_{\ell w}) - u_w(G_{w\ell}) - u_w(C_{w\ell})) \leq 0, \end{aligned}$$

a contradiction. The last inequality follows by (1) and by the fact that α is non-negative.

Now observe that at the final allocation π^*, at most one agent envies the other: if the loser still envies the winner and the winner also envies the loser, then exchanging the bundles would result in a Pareto improvement, a contradiction. Thus, π^* is EF1 when the loser envies the winner at π^*. Consider when at π^* the loser does not envy the winner but the winner envies the loser. Let π' be the previous allocation just before the final transfer, and $X = \pi'(w) \cap \pi^*(w)$ and $Y = \pi'(\ell) \cap \pi^*(\ell)$. By construction, the loser envies the winner more than one item at π', which implies $u_\ell(Y) < u_\ell(X)$. Suppose towards a contradiction that the winner envies the loser more than one item at π^*, which implies $u_w(X) < u_w(Y)$. If g is the last good that has been moved from the winner to the loser, then allocating X to ℓ and $Y \cup \{g\}$ to w would be a Pareto-improvement of π', a contradiction. Similarly, if c is the last chore that has been moved from the loser to the winner, then allocating $X \cup \{c\}$ to ℓ and Y to w would be a Pareto-improvement of π', a contradiction. Hence, the winner envies the loser up to one item; we conclude that π^* is EF1.

It remains to analyze the running time of the algorithm. First, the items can be sorted in $O(m \log m)$ time. The adjustment process takes $O(m^2)$ time. Each iteration checks the allocation is EF1 from the view point of the loser, which requires at most m comparisons of utilities, and there are at most m iterations. Thus, the number of operations is bounded by $O(m^2)$. □

The example below illustrates our discrete adaptation of AW.

Example 9 (Example of the generalized AW)**.** Consider two agents, Alice and Bob, and five items with the following additive utilities where the gray circles correspond to goods and the white circles correspond to chores.

	①	②	③	④	⑤	⑥	⑦
Alice (winner) :	1	-1	2	1	-2	-4	-6
Bob (loser) :	4	-3	6	2	-2	-2	-2
$\lvert u_\ell(o)\rvert/\lvert u_w(o)\rvert$:	4	3	3	2	1	1/2	1/3

The generalized AW initially allocates the goods to Alice and the chores to Bob. Then, it transfers the first good from Alice to Bob and moves the second chore from Bob to Alice. After moving the third good from Alice to Bob, Bob stops being envious up to one item. Hence the final allocation gives the items 2 and 4 to Alice and the rest to Bob.

A natural question is whether PO and EF1 allocation exists for three agents with general additive utilities. We leave this as an interesting open question.

We also note that Pareto-optimality by itself is easy to achieve. We take a permutation of agents and apply a variant of 'serial dictatorship'. The first agents takes all the items for which she has strictly positive utility for. Each subsequent agent does the same for the remaining items. If all agents have been exhausted and there are some items remaining, then the last agents takes all the remaining items.

Proposition 10. *For additive utilities, a Pareto-optimal allocation can be computed in $O(mn)$ time.*

Proof. We take a permutation of agents and apply a variant of 'serial dictatorship'. The first agents takes all the items for which she has strictly positive utility for. Each subsequent agent does the same for remaining items. If all agents have been exhausted and there are some items remaining, then the last agents takes all the remaining items. The resultant allocation is Pareto-optimal. Any allocation that Pareto dominates it will require that the first $n-1$ agents have the same allocation and therefore the last agent has the same allocation as well. □

5. Finding a Connected PROP1 Allocation

We saw that there always exists an EF1 allocation for subjective goods/chores. If we weaken EF1 to PROP1, one can achieve one additional requirement besides fairness, that is, *connectivity*. In this section, we will consider a situation when items are placed on a path, and each agent desires a connected bundle of the path. Finding a connected set of items is important in many scenarios. For example, the items can be a set of rooms in a corridor and the agents could be research groups where each research group wants to get adjacent rooms.

We will show that a connected PROP1 allocation exists and can be found efficiently. In what follows, we assume that the path is given by a sequence of items $(o_1, o_2, \ldots, o_m)$. We first prove a result for the case of the cake cutting setting that is of independent interest. In the following, a *mixed cake* is the interval $[0, m]$. Each agent $i \in N$ has a value density function $\hat{u}_i$, which maps a subinterval of the cake to a real value, where i has uniform utility $u_i(o_j)$ for the interval $[j-1, j]$ for each $j \in [m]$. A *contiguous allocation* of a mixed cake assigns each agent a disjoint sub-interval of the cake where the union of the intervals equals the entire cake $[0, m]$; it satisfies *proportionality* if each agent i gets an interval of value at least the proportional fair share $u_i(O)/n$.

Theorem 11. *For additive utilities, a contiguous proportional allocation of a mixed cake exists and can be computed in polynomial time.*

Proof. Let N^+ be the set of agents with strictly positive total value for O. Now we combine the moving-knife algorithms for goods and chores.

Idea: First, if there is an agent who has positive proportional fair share, i.e., $N^+ \neq \emptyset$, we apply the moving-knife algorithm only to the agents in N^+. Our algorithm moves a knife from left to right, and agents shout whenever the left part of the cake has a value of exactly equal to the proportional fair share. The first agent who shouts is allocated the left bundle, and the algorithm recurs on the remaining instance. Second, if no agent has positive proportional fair share, our algorithm moves a knife from right to left, and agents shout whenever the left part of the cake has value exactly proportional fair share. Again, the first agent who shouts is allocated the left bundle, and the algorithm recurs on the remaining instance. Algorithm 3 formalizes the idea.

Algorithm 3 Generalized Moving-knife Algorithm $\mathcal{A}$

Input: A sub-interval $[\ell, r]$, agent set N', utility functions $\hat{u}_i$ for each $i \in N'$
Output: An allocation π of a mixed cake $[\ell, r]$ to N'

Initialize $\pi(i) = \emptyset$ for each $i \in N'$.
Set $N^+ = \{\, i \in N' \mid \hat{u}_i([\ell, r]) > 0 \,\}$.
if $N^+ \neq \emptyset$ **then**
    if $|N^+| = 1$ **then**
        Allocate $[\ell, r]$ to the unique agent in N^+.
    else
        Let x_i be the minimum point where $\hat{u}_i([\ell, x_i]) = \hat{u}_i([\ell, r])/|N^+|$ for $i \in N^+$.
        Find agent j with minimum x_j among all agents in N^+.
        return π where $\pi(j) = [\ell, x_j]$ and $\pi|_{N' \setminus \{j\}} = \mathcal{A}([x_j, r], N' \setminus \{j\}, (\hat{u}_i)_{i \in N' \setminus \{j\}})$
    end if
else
    Let x_i be the maximum point where $\hat{u}_i([\ell, x_i]) = -\hat{u}_i([\ell, r])/n$ for $i \in N'$.
    Find agent j with maximum x_j among all agents in N'.
    return π where $\pi(j) = [\ell, x_j]$ and $\pi|_{N' \setminus \{j\}} = \mathcal{A}([x_j, r], N' \setminus \{j\}, (\hat{u}_i)_{i \in N' \setminus \{j\}})$
end if

Correctness: We will prove by induction on $|N'|$ that the allocation of a mixed cake produced by $\mathcal{A}$ satisfies the following:

- if $N^+ \neq \emptyset$, then each agent in N^+ receives an interval of value at least the proportional fair share and each agent not in N^+ receives an empty piece; and
- if $N^+ = \emptyset$, then each agent receives an interval of value at least the proportional fair share.

The claim is clearly true when $|N'| = 1$. Suppose that $\mathcal{A}$ returns a proportional allocation of a mixed cake with desired properties when $|N'| = k-1$; we will prove it for $|N'| = k$.

Suppose that some agent has positive proportional fair share, i.e., $N^+ \neq \emptyset$. If $|N^+| = 1$, the claim is trivial; thus assume otherwise. Clearly, agent j receives an interval of value at least the proportional fair share. Further, all other agents in N^+ have value at most the proportional fair share for the left piece $[\ell, x_j]$. Indeed, if there is an agent $i' \in N^+$ whose value for the left piece is greater than the proportional fair share, then i' would have shouted when the knife reaches before x_j by the continuity of $\hat{u}_{i'}$, and $\hat{u}_{i'}([\ell, x_j]) > \hat{u}_{i'}([\ell, \ell])$, contradicting the minimality of x_j. Thus, the remaining agents in N^+ have at least $(n-1) \cdot \hat{u}_i([\ell, r])/n$ value for the rest of the cake $[x_j, r]$; hence, by the induction hypothesis each agent in N^+ gets an interval of value at least the proportional fair share, and each of the remaining agents gets an empty piece.

Suppose that no agent has positive proportional fair share. Again, if there is an agent i' whose value for the left piece $[\ell, x_j]$ is greater than the proportional fair share, then i' would have shouted when the knife reaches before x_j by the continuity of $\hat{u}_{i'}$ and $\hat{u}_{i'}([\ell, x_j]) > \hat{u}_{i'}([\ell, m])$, contradicting the maximality of x_j. Thus, all the remaining agents have value at least $(n-1) \cdot \hat{u}_i([\ell, r])/n$ for the rest of the cake $[x_j, r]$, and hence, by the induction hypothesis, each agent gets an interval of value at least proportional fair share. $\square$

The theorem stated above also applies to a general cake-cutting model in which information about agent's utility function over an interval can be inferred by a series of queries. We note that a contiguous envy-free allocation of a mixed cake is known to exist only when the number of agents is four or a prime number (Segal-Halevi, 2018; Meunier & Zerbib, 2018). Next, we show how a fractional proportional allocation can be used to achieve a contiguous PROP1 division of indivisible items.

Theorem 12. *For additive utilities, a connected PROP1 allocation of a path always exists and can be computed in polynomial time.*

Proof. We know that a fractional contiguous and proportional allocation always exists from Theorem 11. In such an allocation we will not change the allocation of agents who get an empty allocation. As for the rest of the agents we do as follows.

Take any contiguous proportional fractional division of a path. From left to right, we allocate the items on the boundary according to the left-agents preferences. Specifically, we assume without loss of generality that agents $1, 2, \ldots, n$, receive the 1st, 2nd, ..., and n-th bundles. If there is an agent who gets a fraction of one item only under the proportional fractional division, the agent gets nothing under our final allocation. Now suppose that an item o is divided between two agents i and $i+1$. Then we do the following:

1. If two agents disagree on the sign of o, we give the item o to the agent who has positive value for it.

2. If two agents agree on the sign of o, we allocate the item o in such a way that:

    - the left-agent i takes o if the value of o is positive;
    - the right-agent $i+1$ takes o if the value of o is negative.

The resulting integral allocation is PROP1. Clearly, the bundles for the end agents 1 and n satisfy PROP1. Also, the bundles for the middle agents $2, \ldots, n-1$ satisfy PROP1 since such an agent gets value $1/n$ by either receiving the item next to its bundle or deleting the left-most item of his bundle. $\square$

6. Discussion

In this paper, we have formally studied fair allocation when the items are a combination of subjective goods and chores. Our work paves the way for detailed examination of allocation of goods/chores, and opens up an interesting line of research, with many problems left open to explore. In particular, there are further fairness concepts that could be studied from both existence and complexity issues, most notably *envy-freeness up to the least valued item* (EFX) (Caragiannis et al., 2019). In our setting, one can define an allocation π to be *EFX* if for any pair of agents i, j, the following two hold:

1. $\forall o \in \pi(i)$ s.t. $u_i(o) < 0$: $u_i(\pi(i) \setminus \{o\}) \geq u_i(\pi(j))$; and

2. $\forall o \in \pi(j)$ s.t. $u_i(o) > 0$: $u_i(\pi(i)) \geq u_i(\pi(j) \setminus \{o\})$.

That is, i's envy towards j can be eliminated by either removing i's least valuable good from j's bundle or removing i's favorite chore from i's bundle. Caragiannis et al. (2019) mentioned the following 'enigmatic' problem: does an EFX allocation exist for goods? It would be intriguing to investigate the same question for subjective or objective goods/chores.

We also note that while the relationship between Pareto-optimality and most of fairness notions is still unclear, Conitzer et al. (2017) proposed a fairness concept called *Round Robin Share* that can be achieved together with Pareto-optimality. In our context, RRS can be formalized as follows. Given an instance $I = (N, O, U)$, consider the round robin sequence in which all agents have the same utilities as agent i. In that case, the minimum utility achieved by any of the agents is $\text{RRS}_i(I)$. An allocation satisfies RRS if each agent i gets utility at least $\text{RRS}_i(I)$. It would be then very natural to ask what is the computational complexity of finding an allocation satisfying both properties.

Finally, recent papers of Bouveret et al. (2017) and Bilò et al. (2019) showed that a connected allocation satisfying several fairness notions, such as MMS and EF1, is guaranteed to exist for some restricted domains. Although these existence results crucially rely on the fact that the agents have monotonic valuations, it remains open whether similar results can be obtained in fair division of indivisible goods and chores.

Acknowledgements

We would like to thank Bérczi et al. (2020) for pointing out an error of the IJCAI version (Aziz et al., 2019). We thank the reviewers at IJCAI-19 for helpful feedback. Ayumi Igarashi is supported by the KAKENHI Grant-in-Aid for JSPS Fellows number 18J00997. Haris Aziz is supported by a UNSW Scientia Fellowship. Toby Walsh is funded by the European Research Council under the Horizon 2020 Programme via AMPLify 670077.

References

Amanatidis, G., Markakis, E., Nikzad, A., & Saberi, A. (2017). Approximation algorithms for computing maximin share allocations. *ACM Transactions on Algorithms*, *13*(4), 52:1–52:28.

Aziz, H. (2016). Computational social choice: Some current and new directions. In *Proceedings of the 25th International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 4054–4057.

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

Aziz, H., Caragiannis, I., Igarashi, A., & Walsh, T. (2019). Fair allocation of indivisible goods and chores. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*, pp. 53–59.

Aziz, H., Gaspers, S., Mackenzie, S., & Walsh, T. (2015). Fair assignment of indivisible objects under ordinal preferences. *Artificial Intelligence*, *227*, 71–92.

Aziz, H., Rauchecker, G., Schryen, G., & Walsh, T. (2017). Algorithms for max-min share fair allocation of indivisible chores. In *Proceedings of the 31st AAAI Conference on Artificial Intelligence (AAAI)*, pp. 335–341.

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

Barman, S., & Krishnamurthy, S. K. (2020). Approximation algorithms for maximin fair division. *ACM Transactions on Economics and Computation*, *8*(1), 5:1–5:28.

Bérczi, K., Bérczi-Kovács, E. R., Boros, E., Gedefa, F. T., Kamiyama, N., Kavitha, T., Kobayashi, Y., & Makino, K. (2020). Envy-free relaxations for goods, chores, and mixed items..

Bilò, V., Caragiannis, I., Flammini, M., Igarashi, A., Monaco, G., Peters, D., Vinci, C., & Zwicker, W. S. (2019). Almost envy-free allocations with connected bundles. In *Proceedings of the 10th Innovations in Theoretical Computer Science Conference (ITCS)*, pp. 14:1–14:21.

Bogomolnaia, A., Moulin, H., Sandomirskyi, F., & Yanovskaya, E. (2016). Dividing goods and bads under additive utilities. *CoRR*, *abs/1610.03745*.

Bogomolnaia, A., Moulin, H., Sandomirskyi, F., & Yanovskaya, E. (2017). Competitive division of a mixed manna. *CoRR*, *abs/1702.00616*.

Bouveret, S., Cechlárová, K., Elkind, E., Igarashi, A., & Peters, D. (2017). Fair division of a graph. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI 2017, Melbourne, Australia, August 19-25, 2017*, pp. 135–141.

Bouveret, S., Chevaleyre, Y., & Maudet, N. (2016). Fair allocation of indivisible goods. In Brandt, F., Conitzer, V., Endriss, U., Lang, J., & Procaccia, A. D. (Eds.), *Handbook of Computational Social Choice*, chap. 12. Cambridge University Press.

Brams, S. J., & Taylor, A. D. (1996a). *Fair division: From cake-cutting to dispute resolution*. Cambridge University Press.

Brams, S. J., & Taylor, A. D. (1996b). A procedure for divorce settlements. *Issue Mediation Quarterly Mediation Quarterly*, *13*(3), 191–205.

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

Caragiannis, I., Kaklamanis, C., Kanellopoulos, P., & Kyropoulou, M. (2012). The efficiency of fair division. *Theory of Computing Systems*, *50*(4), 589–610.

Caragiannis, I., Kurokawa, D., Moulin, H., Procaccia, A. D., Shah, N., & Wang, J. (2019). The unreasonable fairness of maximum nash welfare. *ACM Transactions on Economics and Computation*, *7*(3), 12:1–12:32.

Conitzer, V., Freeman, R., & Shah, N. (2017). Fair public decision making. In *Proceedings of the 18th ACM Conference on Economics and Computation (ACM-EC)*, pp. 629–646. ACM Press.

de Keijzer, B., Bouveret, S., Klos, T., & 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*, pp. 98–110.

Ghodsi, M., HajiAghayi, M., Seddighin, M., Seddighin, S., & Yami, H. (2018). Fair allocation of indivisible goods: Improvements and generalizations. In *Proceedings of the 19th ACM Conference on Economics and Computation (ACM-EC)*. ACM Press.

Kurokawa, D., Procaccia, A. D., & Wang, J. (2018). Fair enough: Guaranteeing approximate maximin shares. *Journal of the ACM*, *65*(2), 8:1–8:27.

Lipton, R. J., Markakis, E., Mossel, E., & Saberi, A. (2004). On approximately fair allocations of indivisible goods. In *Proceedings of the 5th ACM Conference on Electronic Commerce (ACM-EC)*, pp. 125–131. ACM Press.

Meunier, F., & Zerbib, S. (2018). Envy-free divisions of a partially burnt cake. *CoRR*, *abs/1804.00449*.

Segal-Halevi, E. (2018). Fairly dividing a cake after some parts were burnt in the oven. In *Proceedings of the 17th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pp. 1276–1284. Extended version available as arXiv:1704.00726v5 [math.CO].