

Online Fair Division With Subsidy: When Do Envy-Free Allocations Exist, and at What Cost?

Pooja Kulkarni
Northwestern University
Evanston, IL, United States
pooja.kulkarni@northwestern.edu

Vishnu V. Narayan
University of Illinois Urbana-Champaign
Champaign–Urbana, IL, United States
vishnu.narayan@mail.mcgill.ca

Ruta Mehta
University of Illinois Urbana-Champaign
Champaign–Urbana, IL, United States
rutameht@illinois.edu

Tomasz Ponitka
Tel Aviv University
Tel Aviv, Israel
tomaszp@mail.tau.ac.il

ABSTRACT

We study the problem of fairly allocating m indivisible items arriving online, among n (offline) agents. Although envy-freeness has emerged as the archetypal fairness notion, envy-free (EF) allocations need not exist with indivisible items. To bypass this, a prominent line of research demonstrates that there exist allocations that can be made envy-free by allowing a *subsidy*. Extensive work in the offline setting has focused on finding such *envy-freeable* allocations with bounded subsidy. We extend this literature to an online setting where items arrive one at a time and must be immediately and irrevocably allocated. Our contributions are two-fold:

- **Maintaining EF Online:** We show that envy-freeability cannot always be preserved online when the valuations are submodular or supermodular, even with binary marginals. In contrast, we design online algorithms that maintain envy-freeability at every step for the class of additive valuations, and for its superclasses including k -demand valuations and SPLC valuations.
- **Ensuring Low Subsidy:** We investigate the quantity of subsidy required to guarantee envy-freeness online. Surprisingly, even for additive valuations, the minimum subsidy may be as large as $\Omega(mn)$, in contrast to the offline setting, where the bound is $O(n)$. On the positive side, we identify valuation classes where the minimum subsidy is small (i.e., does not depend on m), including k -valued, rank-one, restricted additive, and identical valuations, and we obtain (mostly) tight subsidy bounds for these classes.

KEYWORDS

Fair Division; Online Algorithms; Subsidy; Envy-Freeness; Resource Allocation

ACM Reference Format:

Pooja Kulkarni, Ruta Mehta, Vishnu V. Narayan, and Tomasz Ponitka. 2026. Online Fair Division With Subsidy: When Do Envy-Free Allocations Exist, and at What Cost?. In *Proc. of the 25th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2026), Paphos, Cyprus, May 25 – 29, 2026*, IFAAMAS, 9 pages. <https://doi.org/10.65109/DXUE6939>



This work is licensed under a Creative Commons Attribution International 4.0 License.

Proc. of the 25th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2026), C. Amato, L. Dennis, V. Mascardi, J. Thangarajah (eds.), May 25 – 29, 2026, Paphos, Cyprus. © 2026 International Foundation for Autonomous Agents and Multiagent Systems (www.ifaamas.org). <https://doi.org/10.65109/DXUE6939>

1 INTRODUCTION

In the decades since its formal introduction by Steinhaus [44] in 1948, the fair division problem has emerged as a fundamental topic at the intersection of economics and computer science. The central fair division question asks how to allocate a finite collection of items amongst a set of agents, each equipped with a valuation function over subsets of the items, in a manner that is universally regarded as fair. When the items are divisible, and with only mild assumptions on the structure of the agents' valuations, it is possible to obtain an allocation where no agent is envious of any other, i.e., every agent (according to its own valuation) weakly prefers its assigned bundle to that of any other agent [45]. This concept, introduced by Foley [23], is now widely known as *envy-freeness*. However, when the items are indivisible and must each be integrally assigned to some agent, it is easy to see that envy-free allocations do not necessarily exist: when only one valuable item is available, the agent that receives it is always envied.

In a seminal work, Maskin [41] asked whether this impossibility could be circumvented by the addition of a single divisible good (i.e., money), and if so, how much of it was needed to eliminate all envy. Maskin [41] showed that for matching instances with n agents and $m = n$ goods, if without loss of generality any specific good has a value of at most one dollar, an *envy-freeable* allocation always exists, and a subsidy of $n - 1$ dollars is sufficient to remove all envy for some allocation. This result spurred a large body of literature investigating the existence of such envy-freeable allocations in fair division [6, 13, 28, 31, 32], leading to a now well-known characterization:

THEOREM [6, 28, 31, 32]. *An allocation is envy-freeable if and only if it is locally efficient.*

An allocation is locally efficient (LE) if, when the n assigned bundles of the allocation are each frozen, no reassignment of these bundles to the n agents increases the social welfare. Several papers have since examined the quantity of subsidy sufficient to guarantee envy-freeness. Prominently, the work of Brustle et al. [13] proves that it is always possible to eliminate envy with a *small* total subsidy, i.e., one that does not grow with the number of items m .

We extend the above body of literature to a natural online setting in which items arrive one at a time, revealing their values, and must each be assigned immediately and irrevocably. Our interest in this problem arises from the simple observation that when the valuations are additive, it is possible to *maximize welfare* online

(i.e., maintain a kind of *global* efficiency), implying the maintenance of LE and therefore envy-freeability online. Thus the first natural question that we raise and study is the following.

Question 1. *Going beyond additive valuations, for what valuation classes is it possible to maintain envy-freeability online?*

We show that it is additionally possible to maintain LE online for the classes of SPLC valuations and k -demand valuations (for any k). Interestingly, although welfare maximization is one possible strategy to obtain LE, the collection of valuation classes for which LE can be maintained online is strictly more general, i.e., this may be possible even when maximizing welfare online is impossible. In this sense, our work demonstrates a separation between classes for which online welfare maximization is possible and classes for which local efficiency can be maintained online.

In the other direction, we show that maintaining LE online is *impossible* for budget-additive valuations, even if the budgets are known in advance, and that it is impossible for submodular and supermodular valuations, even if the marginal values are binary.

While in the offline case, envy-freeness with a small subsidy is feasible for both additive valuations and monotone valuations, for the online problem, maintaining LE via welfare maximization results in a subsidy that grows linearly with m in the worst case. This raises the second natural question that we study in this work.

Question 2. *Do there exist valuation classes for which it is possible to maintain envy-freeability online with a small subsidy, i.e., one that does not grow with m ?*

We first show that when the agents have additive valuations this is impossible, and there exists an instance for which maintaining envy-freeability online necessitates a subsidy as high as $m(n - 1)$. A similar lower bound extends to SPLC valuations and k -demand valuations (both of which generalize additive valuations when k is allowed to be as large as m).

In contrast, we provide a positive answer to the above question for several well-studied valuation classes, including unit-demand and bivalued valuations (and, more generally, k -demand and k -valued valuations when k is independent of m), as well as rank-one, restricted additive, and identical monotone valuations. The definitions of these classes, along with a more detailed discussion of their properties, are presented in the respective sections.

All of our positive results are obtained algorithmically. Through adversarial constructions, we show that nearly all of our algorithms are *exactly* tight, i.e., they have the best possible bound on the total subsidy even up to coefficients. Interestingly, for several of these classes this bound is asymptotically larger (as a function of n) than the corresponding bound in the offline version of the problem, thereby providing strong separations between the online and offline settings.¹ Table 1 shows a high-level summary of our results.

We remark that a key modeling choice in our work is to track the minimum subsidy required to guarantee envy-freeness at each step, rather than maintaining an explicit online update of the payment vector. At any point during the online arrival of items, the mechanism is prepared to obtain an envy-free allocation with subsidy should the process terminate. This choice aligns with applications

¹For instance, a result of Brustle et al. [13] shows that a total subsidy of $n - 1$ suffices for additive valuations. Our results demonstrate that the minimum subsidy is $\Theta(n^2)$ even for structured subclasses of additive valuations, including binary additive valuations and rank-one valuations.

Valuation Class	Question 1 Can EF be maintained online?	Question 2 Can we get EF with small subsidy?
Budget Additive	✗ (Thm 3.4)	✗ (—)
Binary Submodular	✗ (Thm 3.5)	✗ (—)
Binary Supermodular	✗ (Thm 3.6)	✗ (—)
SPLC	✓ (Thm 3.2)	✗ (—)
Additive	✓ (—)	✗ (Thm 4.1)
k -Demand	✓ (—)	✓* (Thm 4.2)
k -Valued	✓ (—)	✓* (Thm 4.3)
Rank-One	✓ (—)	✓ (Thm 4.4)
Restricted Additive	✓ (—)	✓ (Thm 4.5)
Identical Monotone	✓ (—)	✓ (Thm 4.6)

Table 1: Summary of our results for Questions 1 and 2 for the valuation classes considered in this work. (*) For k -valued and k -demand valuations, our subsidy bound is small in the sense that it does not depend on m but does depend on k . For the rows marked with “(—)” in Question 1, the positive result is implied by Theorem 3.2, while similarly for Question 2, the negative result is implied by Theorem 4.1.

in which the online arrival of items may halt unpredictably, and the envy-freeness guarantee must be immediately realized with bounded subsidy.

Finally, we note that most of our results extend to the related chore-division problem, where each item adds a *cost* to the agent that receives it. Section 5 contains a high-level discussion of the problem of bounding the subsidy in the fair division of chores.

The full version of this work is available in [36].

1.1 Additional Related Work

While this work introduces the online fair division with subsidy problem with online items and offline agents, there is a large body of research on each of these topics taken individually (see Amanatidis, Aziz, Birmpas, Filos-Ratsikas, Li, Moulin, Voudouris, and Wu [3]). *Fair Division With Subsidy.* The problem of finding a fair allocation with subsidy has been considered in several contexts, including rent division [21, 24], item allocation [6, 7, 9, 13, 16, 31, 35, 38, 46, 47] and job scheduling [19, 22, 32], with a variety of objectives including bounding payments, measuring social welfare, capping makespan, and approximating minimum subsidy.

Online Fair Division. The problem of finding fair allocations in online settings has been extensively explored in the literature on online fair division. Prior work has examined various formulations of the problem, including settings with online items and offline agents [27, 30, 33], offline items and online agents [34, 37, 48], as well as randomized mechanisms [2, 12]. Primary research objectives include minimizing the maximum envy, bounding the number of online adjustments, approximating social welfare, and approximating the maximin share.

2 PRELIMINARIES

We consider a fair division setting with a set of agents $[n] = \{1, 2, \dots, n\}$ and a set of items $[m] = \{1, 2, \dots, m\}$. Each agent $i \in [n]$ is equipped with a valuation function $v_i : 2^{[m]} \rightarrow \mathbb{R}_{\geq 0}$. We make the standard assumptions that each valuation is monotone, i.e. $v_i(S) \leq v_i(T)$ for all $S \subseteq T \subseteq [m]$, and normalized, i.e. $v_i(\emptyset) = 0$. Our theorems are written for the setting where the items are goods and the marginal values are therefore nonnegative, but, as we discuss later, a similar analysis can be applied to the case of chores without much further effort.

We consider some important types of instances with additional assumptions on the structure of the valuations, which we define in the corresponding subsections.

Our task is to select an allocation $X = (X_1, X_2, \dots, X_n)$ and a subsidy vector $p = (p_1, \dots, p_n)$. An allocation is a partition of $[m]$ into n bundles such that $X_1 \cup X_2 \cup \dots \cup X_n = [m]$ and $X_i \cap X_j = \emptyset$ for $i \neq j$. A subsidy vector must satisfy $p_i \geq 0$ for all $i \in [n]$.

2.1 Fairness Notions and Local Efficiency

Envy-freeness, a central fairness notion in the fair division literature requires that no agent envies the bundle of another agent. An allocation $X = (X_1, \dots, X_n)$ is envy-free (EF) if $v_i(X_i) \geq v_i(X_j)$ for all $i, j \in [n]$. Since envy-free allocations do not necessarily exist, we need the use of a subsidy.

Definition 2.1 (Envy-Freeness with Subsidy, Envy-Freeability). An allocation $X = (X_1, \dots, X_n)$ together with a subsidy vector $p = (p_1, \dots, p_n)$ is envy-free if $v_i(X_i) + p_i \geq v_i(X_j) + p_j$ for all $i, j \in [n]$. An allocation X is said to be envy-freeable if a subsidy vector satisfying this condition exists.

The social welfare of an allocation is the sum of the values obtained by all agents, i.e., $SW(X) = \sum_{i \in [n]} v_i(X_i)$. A well-known result characterizes envy-freeness via local efficiency.

Definition 2.2 (Local Efficiency). An allocation $X = (X_1, \dots, X_n)$ is locally efficient if there is no permutation of the agents that increases the social welfare, i.e., X is LE if for all $\pi : [n] \rightarrow [n]$ we have $\sum_{i \in [n]} v_i(X_i) \geq \sum_{i \in [n]} v_{\pi(i)}(X_i)$.

THEOREM 2.3 ([6, 28, 31, 32]). *An allocation $X = (X_1, \dots, X_n)$ is envy-freeable if and only if X is locally efficient.*

Clearly, any allocation that maximizes the social welfare (across all possible allocations) is locally efficient.

2.2 Online Setting

We consider an online setting where items $1, 2, \dots, j$ arrive one by one and must be irrevocably allocated upon arrival. This means that when item j arrives, the items $1, 2, \dots, j-1$ are already allocated to some agents, and the algorithm must allocate item j before item $j+1$ arrives. We assume that the online algorithm has value-oracle access, meaning that when item j arrives, the algorithm can access any agent's value for any bundle containing the items that have arrived up to that point, i.e., the algorithm can query $v_i(S)$ for any $i \in [n]$ and $S \subseteq \{1, \dots, j\}$. We further assume that the algorithm selects a subsidy vector $p = (p_1, \dots, p_n)$ after all items have been

allocated, and we aim to construct algorithms that ensure envy-freeness with subsidy (Definition 2.1) at the end of the execution, once all items have arrived.

Note that in our online model, the algorithm does not know the number of items in advance. Any algorithm that guarantees envy-freeness for online instances must maintain local efficiency at every step, i.e., whenever the online arrival of items (unpredictably) terminates, the algorithm must output a locally efficient allocation and a subsidy vector that eliminates all envy. Consequently, rather than updating the subsidies online, the algorithm must only maintain a locally efficient allocation online, along with a minimum envy-eliminating payment vector for that allocation.

2.3 Minimum Subsidy and the Envy Graph

For classes of valuations where envy-freeness with subsidies is attainable, our goal is to design online algorithms that use as little total subsidy as possible, formally defined below.²

Definition 2.4 (Total Subsidy). A subsidy vector $p = (p_1, \dots, p_n)$ is said to use total subsidy at most X if $\sum_{i \in [n]} p_i \leq X$.

Throughout this paper, as in prior work on bounding the subsidy, we assume that the marginal value of any item is at most 1, i.e., for any $S \subseteq [m]$ and $g \in [m]$, $v_i(S \cup \{g\}) - v_i(S) \leq 1$. This assumption can be made without loss of generality by appropriately scaling all valuations by the same factor, which only changes the relative units in which the subsidy is measured.

For a *given* envy-freeable allocation, it was shown by [31] that an envy-eliminating subsidy vector that minimizes the subsidy for each agent (and therefore also minimizes the total subsidy) can be found in polynomial time. In this work, rather than bounding the per-agent subsidy, we focus on bounding the total subsidy, which is at most n times larger. From the perspective of our notion that a *small* subsidy is independent of the number of items, the two objectives are essentially equivalent.

For the analysis of fair allocations with subsidies, we will utilize an auxiliary directed graph defined for every allocation called the *envy graph*. For a given fair division instance, and given allocation X , the envy graph G_X is constructed as follows. The agents $[n]$ form the set of vertices of this graph, and there is a weighted directed arc for every ordered pair of agents. The weight on arc (i, k) is the envy that agent i has for agent k in X , i.e., $w_{G_X}(i, k) = v_i(X_k) - v_i(X_i)$.

Halpern and Shah [31] observed that if an allocation is locally efficient, then the corresponding envy graph does not have any directed cycles of positive weight. Thus the heaviest path starting at any vertex $i \in V(G_X)$ is well-defined. An important lemma for bounding the total subsidy, proved in [31] for additive valuations and in [13] for monotone valuations, is as follows.

LEMMA 2.5 ([13, 31]). *Given a fair division instance with m goods, n agents, and monotone valuation functions v_i , consider an allocation $X = (X_1, \dots, X_n)$ that is locally efficient. Let $\ell(i)$ denote the weight of the heaviest path in the envy graph G_X starting at vertex i . Then the minimum subsidy required to make X envy-free is exactly $\sum_{i \in [n]} \ell(i)$. Moreover, this quantity can be bounded above by $m(n-1)$.*

²Note that a simple reduction from the Partition problem shows that deciding whether a given instance has an envy-free allocation is NP-hard. Thus the optimization problem of minimizing the subsidy for a given instance is also NP-hard.

3 BOUNDARIES OF ENVY-FREEABILITY

Recall that in order to maintain envy-freeability, we are tasked with maintaining local efficiency online; that is, after each incoming item is assigned to some agent, we require that no permutation of the bundles increases the social welfare of the resulting allocation.

We begin our analysis with additive valuations, which admit an interesting property in the online setting: when a new item arrives, for any locally efficient partial allocation (of the items that have previously arrived), there exists an agent to which the new item can be assigned such that the resulting allocation is locally efficient. In other words, local efficiency can always be extended online.

PROPOSITION 3.1. *In a fair division instance with additive valuations, let $S \subseteq [m]$ be a subset of items, and let $g \in [m] \setminus S$ be an item not in S . Given any partial allocation X of the items in S that is locally efficient, let X' be the allocation obtained by assigning g to some agent $i^* \in \arg \max_{i \in [n]} v_i(\{g\})$. Then X' is locally efficient.*

PROOF. Suppose, for a contradiction, that there exists a permutation $\pi : [n] \rightarrow [n]$ such that reallocating the bundles of X' along π increases the social welfare. Then, since the valuations are additive,

$$\begin{aligned} & \sum_{i \in [n]} v_{\pi(i)}(X'_i) > \sum_{i \in [n]} v_i(X'_i) \\ \implies & v_{\pi(i^*)}(X'_{i^*} \setminus \{g\}) + \sum_{i \in [n] \setminus \{i^*\}} v_{\pi(i)}(X'_i) \\ & > v_{i^*}(X'_{i^*} \setminus \{g\}) + \sum_{i \in [n] \setminus \{i^*\}} v_i(X'_i) \quad (1) \\ \implies & \sum_{i \in [n]} v_{\pi(i)}(X_i) > \sum_{i \in [n]} v_i(X_i), \end{aligned}$$

which contradicts the fact that X is locally efficient. \square

The above proposition immediately gives us a simple algorithm for maintaining local efficiency online when valuations are additive: assign each item to an agent that has the highest value for it, thereby maximizing the social welfare. Observe that if the valuations are not additive, the implication in (1) is no longer necessarily true. This raises the natural question of whether local efficiency can be maintained online for classes beyond additive valuations. We answer this question either affirmatively or negatively for a variety of valuation classes. We begin with our positive results.

3.1 LE Can Be Maintained Online

We show that LE can be maintained online for the class of SPLC valuations, and the class of k -demand valuations for any k (i.e., we allow k to be as large as m). These valuation classes each generalize the class of additive valuations.

3.1.1 SPLC Valuations. A valuation function v_i is SPLC (Separable Piecewise-Linear Concave), if the set of goods can be partitioned into T types such that any two items of the same type are identical, for type $t \in [T]$ there is a set M_t of goods of that type, and the agent has a value $v_{i,\ell}^t$ associated with the ℓ^{th} copy of any good of type t . Across different types, the valuation function is additive. The function is concave, so that $v_{i,1}^t \geq v_{i,2}^t \geq \dots \geq v_{i,|M_t|}^t$. Putting this all together, the value for a given bundle S is $v_i(S) = \sum_{t \in [T]} \sum_{\ell \in [|S \cap M_t|]} v_{i,\ell}^t$. These valuations, also called additively

separable concave valuations, have been previously studied in the context of fair division [e.g. 5, 17, 18].

THEOREM 3.2. *Envy-freeability can be maintained online for all instances with SPLC valuations.*

PROOF. The algorithm for maintaining LE in this case is simple: it assigns each incoming good to the agent with the highest marginal value for it. We show that, even for SPLC valuations, this algorithm finds a social welfare maximizing allocation and is therefore locally efficient. The social welfare of an allocation $X = (X_1, \dots, X_n)$ is

$$\begin{aligned} SW(X) &= \sum_{i \in [n]} v_i(X_i) = \sum_{i \in [n]} \sum_{t \in [T]} \sum_{\ell \in [|S \cap M_t|]} v_{i,\ell}^t \\ &= \sum_{t \in [T]} \sum_{i \in [n]} \sum_{\ell \in [|S \cap M_t|]} v_{i,\ell}^t \end{aligned}$$

where we swap the order of summation in the last equality. To prove that our allocation maximizes welfare, it therefore suffices to prove that for each type $t \in [T]$, it maximizes $\sum_{i \in [n]} \sum_{\ell \in [|S \cap M_t|]} v_{i,\ell}^t$.

Now, fix a type $t \in [T]$. This type has $|M_t|$ copies in the instance, so each agent has a set of $|M_t|$ marginal values for different copies of this good for a total of at most $n|M_t|$ possible marginals. These valuations are concave i.e., each new copy has a decreasing marginal value for any agent. Since the algorithm allocates an incoming copy to an agent with the highest marginal, it chooses the highest $|M_t|$ marginal values for allocating this good, which maximizes the social welfare amongst possible assignments. \square

3.1.2 k -Demand Valuations. A valuation function v_i belongs to the class of k -demand valuations if the value of any bundle S is equal to the sum of the values of the k highest-valued items, valued as singletons, in S . That is, every item $j \in [m]$ has an associated value $v_i(\{j\})$, and the value of a non-empty bundle S is $v_i(S) = \max_{T \subseteq S: |T| \leq k} \sum_{j \in T} v_i(\{j\})$. This class notably generalizes unit-demand valuations (when $k = 1$) and additive valuations (when $k \geq m$) and models scenarios where an agent can hold at most k items; see [e.g. 15, 20, 50]. Interestingly, unlike the case of additive or SPLC valuations, for k -demand valuations it is impossible to maintain a social-welfare-maximizing allocation online, as shown in the following example. Consider an instance with unit-demand valuations, with $n = 2$ agents and $m = 2$ items, where agent 1 has values 0.75 and 1 respectively for the two items, and agent 2 has values 0.5 and 0.25 respectively for the two items. The welfare-maximizing allocation for this instance gives the second item to agent 1 and the first item to agent 2, for a total value of 1.5. However, in the online setting, when item 1 arrives, it is assigned to agent 1 (since $0.75 > 0.5$) to maximize welfare. Nonetheless, we show that for k -demand valuations it is possible to maintain local efficiency (and envy-freeability) in the online setting.

THEOREM 3.3. *Envy-freeability can be maintained online for all instances with k -demand valuations.*

PROOF. Consider the following algorithm. Each time a good g arrives, the algorithm allocates it to the agent i_g with the highest singleton value for it, i.e., $i_g \in \arg \max_{i \in [n]} v_i(\{g\})$. Let the resulting allocation be $X = (X_1, \dots, X_n)$. When the agents have k -demand valuations, at every step in the execution of the online algorithm, this allocation ensures that each bundle is assigned to the agent with

the highest value for this bundle, i.e., $v_i(X_i) \geq v_{i'}(X_i)$ for all $i, i' \in [n]$. To see this, fix any pair of agents $i, i' \in [n]$. By the definition of k -demand valuations, $v_{i'}(X_i) = \max_{T \subseteq X_i: |T| \leq k} \sum_{j \in T} v_{i'}(\{j\})$. Let T^* be a subset of X_i that achieves this maximum. Then $v_{i'}(X_i) = \sum_{j \in T^*} v_{i'}(\{j\})$, which by our algorithm is at most $\sum_{j \in T^*} v_i(\{j\}) \leq v_i(X_i)$, where the last inequality follows from the definition of k -demand valuations. Thus $v_i(X_i) \geq v_{i'}(X_i)$ for all $i, i' \in [n]$, so there is no reassignment of the n fixed bundles that increases the welfare. Consequently, the algorithm maintains LE. \square

3.2 LE Cannot Be Maintained Online

In this section, we show that it is impossible to maintain local efficiency online for budget-additive valuations, even if the budgets are known in advance, and for submodular and supermodular valuations, even if the marginal values are binary (i.e., in $\{0, 1\}$). The impossibilities for binary submodular and binary supermodular valuations are particularly surprising, given the large collection of positive results in fair division for these highly structured valuation classes [e.g. 8, 10, 11].

3.2.1 Budget-Additive Valuations. A valuation function v_i is budget-additive if it is additive, but capped at an agent-specific budget B_i . The value of a non-empty bundle S is $v_i(S) = \min\{\sum_{j \in S} v_i(\{j\}), B_i\}$. Budget-additive valuations have been widely studied in auctions and fair division [e.g. 14, 25, 42].

THEOREM 3.4. *No online algorithm can maintain envy-freeability in instances with budget-additive valuations, even if the budgets are known in advance.*

PROOF. Consider an instance with two agents where agent 1's budget is $1 - \epsilon$, and agent 2's budget is 1. Suppose an item arrives online with value $1 - \epsilon$ for agent 1 and value $1 - 2\epsilon$ for agent 2. Any online algorithm that maintains LE assigns this item to agent 1. Next, suppose a second item arrives online with value $1 - \epsilon$ for agent 1 and value $1/2$ for agent 2. If this item is assigned to agent 1, the resulting social welfare is $1 - \epsilon$, but swapping the two bundles results in an increase in the social welfare to 1. Similarly, if the second item is instead assigned to agent 2, the resulting social welfare is $3/2 - \epsilon$, but swapping the two bundles results in a social welfare of $2 - 3\epsilon$, which is an increase for sufficiently small $\epsilon > 0$. \square

3.2.2 Binary Submodular Valuations. Next, we show that local efficiency cannot be maintained online for the class of submodular valuations. This result holds even in the special case where the marginal value of an item is either 0 or 1, i.e., submodular valuations with binary marginals, and therefore extends to matroid-rank valuations and gross-substitutes valuations [40]. It is well-known that the class of submodular valuations with binary marginals exactly coincides with the class of matroid-rank valuations.

A valuation function v_i is matroid rank if there is an underlying finite matroid $\mathcal{M}_i = ([m], \mathcal{I}_i)$ such that the value of a bundle $S \subseteq [m]$ is equal to the rank of that bundle in \mathcal{M}_i , i.e., the size of the largest independent subset of S . In the following proof, we will use the fact that matroids are closed under restriction, i.e., if an online instance with a given profile of matroid-rank valuations terminates early, then the resulting partial instance is a feasible instance for the class of matroid-rank valuations.

THEOREM 3.5. *No online algorithm can maintain envy-freeability in instances with binary submodular valuations.*

PROOF. For a given online algorithm, we will adversarially construct an instance with matroid-rank valuations for which this algorithm cannot maintain local efficiency online. The instance has two agents, labeled 1 and 2, and up to four items, labeled a, b, c and d , that arrive in order. Suppose both agents have a value of 1 for each of the items a and b and value 2 for the bundle $\{a, b\}$. Without loss of generality, suppose the algorithm assigns item a to agent 1.

When item b arrives, we have the following cases.

CASE 1: The algorithm assigns item b to agent 2. Suppose item c arrives next, and agent 1's underlying matroid has the bases $\{\{a, b\}, \{b, c\}\}$, and agent 2's matroid has the bases $\{\{a, b\}, \{a, c\}\}$ (this choice is consistent with the valuations of a and b). If item c is assigned to agent 1, then the social welfare of the resulting allocation is 2. However, swapping the bundles increases the social welfare to 3. Similarly, if item c is assigned to agent 2, then swapping the bundles increases the social welfare from 2 to 3. Consequently, the online algorithm fails to maintain local efficiency.

CASE 2: The algorithm assigns item b to agent 1. We will now require item d to complete the argument. Suppose agent 1's underlying matroid consists of the bases $\{\{a, b\}, \{a, c\}, \{a, d\}, \{b, c\}, \{b, d\}, \{c, d\}\}$, and agent 2's underlying matroid has the bases $\{\{a, b, c\}, \{a, b, d\}\}$ (these choices are consistent with valuations of a and b). We have two cases, corresponding to the possible assignments of item c .

Case 2a: The algorithm assigns item c to agent 1. If item c is assigned to agent 1, then the social welfare of the resulting allocation is 2. However, swapping the bundles increases the social welfare to 3, so the online algorithm fails to maintain local efficiency in the restricted instance.

Case 2b: The algorithm assigns item c to agent 2. Suppose item c is assigned to agent 2. Then, when item d appears, it is assigned to either agent 1 or agent 2. If d is assigned to agent 1, then the social welfare of the resulting allocation is 3. However, swapping the bundles increases the social welfare to 4, so the online algorithm fails to maintain local efficiency. For the other case, suppose d is assigned to agent 2, then the social welfare of the resulting allocation is again 3. However, swapping the bundles once again increases the social welfare to 4, so the online algorithm fails to maintain local efficiency in all cases. \square

3.2.3 Binary Supermodular Valuations. We present a similar result for binary supermodular valuations. The argument has a more complicated case analysis that also requires the use of a fifth item e . Due to length restrictions, the proof of this result is deferred to the full version [36].

THEOREM 3.6. *No online algorithm can maintain envy-freeability in instances with binary supermodular valuations.*

We remark that it is much easier to construct small counterexamples for the classes of submodular and supermodular valuations if the assumption of binary marginal values is dropped.

4 (MOSTLY) TIGHT SUBSIDY BOUNDS

In the previous section, our goal was to delineate the valuation classes for which it is possible to maintain local efficiency online.

For this section, our objective is to maintain local efficiency while additionally bounding the total subsidy sufficient for envy-freeness in the resulting allocation. We investigate a variety of valuation classes in order to determine tight upper bounds and lower bounds on the subsidy for these classes.

As a consequence of Lemma 2.5 the upper bound of $m(n-1)$ holds for all valuation functions that we consider in this section. First, in Section 4.1, we show that for additive valuations (and therefore its superclasses, including SPLC valuations and k -demand valuations for unrestricted k), this is the best subsidy bound that any online algorithm can achieve. Next, in Section 4.2 we investigate structured classes of valuations and show that this subsidy bound improves. In particular, we show that a *small* subsidy, independent of the number of items m , is possible for these classes. Most of our subsidy bounds in this section are exactly tight even up to coefficients. The proofs of some of our theorems are omitted from this version and appear in the full version [36].

4.1 Large Subsidy Bounds

In this section, we consider the case of additive and, more generally, SPLC valuations, as discussed in Section 3.1.1.

THEOREM 4.1. *There exists an online algorithm that achieves envy-freeness with a total subsidy of at most $m(n-1)$ for all instances with SPLC valuations, which in particular include all additive valuation instances. Moreover, for any $\epsilon > 0$ and any online algorithm, there exists an additive instance for which achieving envy-freeness requires a total subsidy of at least $m(n-1) - \epsilon$.*

4.2 Small Subsidy Bounds

In this section, we consider five well-studied valuation classes for which we establish small subsidy bounds. These upper and lower bounds are exactly tight, except for the case of k -valued valuations.

4.2.1 k -Demand Valuations. We begin by analyzing the case of k -demand valuations, as discussed in Section 3.1.2.

THEOREM 4.2. *There exists an online algorithm that achieves envy-freeness with a total subsidy of at most $k(n-1)$ for all instances with k -demand valuations. Moreover, for any $\epsilon > 0$ and any online algorithm, there exists a k -demand instance for which achieving envy-freeness requires a total subsidy of at least $k(n-1) - \epsilon$.*

Our proof of the upper bound employs an algorithm that assigns each item to the agent that values it the most. Notably, this allocation rule does not necessarily maximize welfare for this class of valuations, yet we show it preserves local efficiency.

4.2.2 Personalized k -Valued Valuations. A valuation function v_i is k -valued if there exist a_1, \dots, a_k such that v_i is additive and $v_i(\{g\}) \in \{a_1, \dots, a_k\}$ [e.g. 26]. An important special case is the *bivalued* case, corresponding to $k = 2$ [e.g. 4]. Our results apply to the setting of *personalized k -valued valuations*, where the set of values $\{a_1, \dots, a_k\}$ may differ across agents.

THEOREM 4.3. *There exists an online algorithm that achieves envy-freeness with a total subsidy of at most $n^2 k^n$ for all instances with personalized k -valued valuations.*

PROOF. In personalized k -valued valuations, each good can take at most k different values for each agent. Therefore, if we represent each good as an n -dimensional vector where the i^{th} coordinate denotes the value that agent i assigns to the good, the set of all possible goods can be represented by at most k^n distinct vectors. We refer to each of these vectors as a *type*.

The algorithm maps every good to its corresponding type. The first good of a type is allocated to the agent who values it the most, the second to the agent who values it second-highest, and so on in a round-robin fashion, breaking ties arbitrarily.

We first show that the resulting allocation is locally efficient. Suppose, for a contradiction, that the envy graph has a positive-weight directed cycle (corresponding to a violation of local efficiency). Then, since the allocation can be additively decomposed into k^n separate allocations, one corresponding to each type, there exists an item type such that the allocation of the items of that type violates local efficiency. However, within each type, the items are assigned in non-increasing order of agent values, thus any agent that has a greater value for some item type receives a (weakly) larger number of items of that type, so by the rearrangement inequality [49] the allocation within each type is locally efficient.

It remains to bound the subsidy for the above algorithm. For any given type, the envy between any two agents is bounded by the value of a single good. Consequently, the total envy between any two agents across all types is bounded by the total number of types, i.e., k^n . Thus the weight of the heaviest path starting at any vertex is at most $(n-1)k^n$, and by Lemma 2.5, the total subsidy is bounded by $n^2 k^n$. \square

4.2.3 Rank-One Valuations. An instance with valuation functions v_1, \dots, v_n is *rank-one* if $v_i(X) = \sum_{j \in X} w_j \cdot q_j$ for some agent weights $w_1, \dots, w_n \in [0, 1]$ and base values $q_1, \dots, q_m \in [0, 1]$. An important motivation for rank-one valuations arises in the sponsored search model, where agents correspond to advertisers and items to ad slots. Here, an advertiser's value for a slot is the product of their value per click and the slot's click-through rate [39].

THEOREM 4.4. *There exists an online algorithm that achieves envy-freeness with a total subsidy of at most $n(n+1)/2 - 1$ for all instances with rank-one valuations. Moreover, for any $\epsilon > 0$ and any online algorithm, there exists a rank-one instance for which achieving envy-freeness requires a total subsidy of at least $n(n+1)/2 - 1 - \epsilon$.*

PROOF. Throughout the proof, we denote $q(S) = \sum_{g \in S} q_g$.

Upper bound. Let X denote the current allocation and g be the arriving item. The algorithm maintains the invariant $q(X_1) \geq q(X_2) \geq \dots \geq q(X_n)$, which implies local efficiency by the rearrangement inequality [49]. The allocation rule for each arriving item g is defined as follows. If there exists an agent $i \in [n-1]$ such that $q(X_i) \geq q(X_{i+1}) + 1$, then the algorithm allocates g to agent $i+1$. If no such agent exists, the algorithm allocates g to agent 1.

We first observe that, by construction, at any point in the execution of the algorithm there can be at most one index i such that $q(X_i) \geq q(X_{i+1}) + 1$. Moreover, for agent i , $q(X_i) \leq q(X_{i+1}) + 2$.

By the characterization of the minimum subsidy (Lemma 2.5), the subsidy required to make the allocation envy-free is equal to the weight of the heaviest path starting at i in the envy graph.

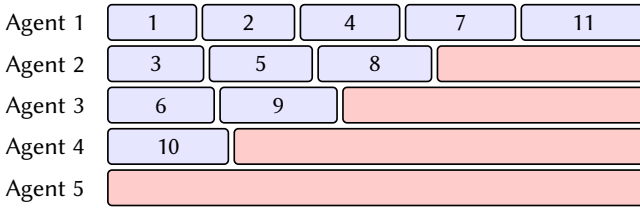


Figure 1: Illustration of the lower bound construction for rank-one valuations in the proof of Theorem 4.4. Each blue bar represents an item, with its length proportional to the item’s base value and the label inside indicating its arrival order. The allocation shown is the subsidy-minimizing one that maintains envy-freenability at every step. The red bar lengths are proportional to each agent’s envy toward agent 1.

The weight of a path $i_1 \rightarrow i_2 \rightarrow \dots \rightarrow i_k$ is given by $\sum_{r=1}^{k-1} w_r \cdot (q(X_{r+1}) - q(X_r))$.

We now argue that, without loss of generality, the heaviest path can be assumed to have the form $i \rightarrow i - 1 \rightarrow \dots \rightarrow 1$. Indeed, consider any arc $i \rightarrow j$ with $i > j$. This edge can be replaced by the path $i \rightarrow i - 1 \rightarrow \dots \rightarrow j + 1 \rightarrow j$, whose weight is $\sum_{r=j+1}^i w_r \cdot (q(X_{r-1}) - q(X_r)) \geq \sum_{r=j+1}^i w_i \cdot (q(X_{r-1}) - q(X_r)) = w_i \cdot (q(X_i) - q(X_j))$. Thus, this replacement cannot decrease the path weight. Furthermore, since the envy graph contains no positive-weight cycle by Lemma 2.5, the heaviest path is simple. Combined with the invariant $q(X_1) \geq q(X_2) \geq \dots \geq q(X_n)$, this implies that the heaviest path indeed has the structure $i \rightarrow i - 1 \rightarrow \dots \rightarrow 1$.

For such a path, the subsidy required for agent i is at most $\sum_{r=2}^i w_r \cdot (q(X_{r-1}) - q(X_r)) \leq \sum_{r=2}^i (q(X_{r-1}) - q(X_r)) = q(X_1) - q(X_i) \leq i$ where the final inequality follows from the observation above. Thus each agent i with $2 \leq i \leq n$ receives a subsidy of at most i , and the total subsidy is bounded by $2 + 3 + \dots + n = n(n+1)/2 - 1$.

Lower bound. Fix any online algorithm for the class of rank-one valuations. Consider the following rank-one instance with n agents and $m = n(n+1)/2$ goods. The agents have parameters $w_i = 1 - i \cdot \epsilon$ for $i \in [n]$, and the goods have parameters $q_j = 1 - \epsilon + 2^{j-n} \cdot \epsilon$ for $j \in [m]$, where $0 < \epsilon < 1/n$.

We first claim that for any $i \in [n - 1]$, throughout the execution of the algorithm, either both agents i and $i + 1$ are assigned no goods, or agent i is assigned strictly more goods than agent $i + 1$. It is easy to see that agent $i + 1$ can never receive strictly more goods than agent i , because $w_{i+1} < w_i$, and swapping their bundles in such a situation strictly increases welfare, contradicting local efficiency. Suppose that at some point agent $i + 1$ is assigned an item j , and at that moment both agents i and $i + 1$ have exactly k goods. Let X_i and X_{i+1} be their respective bundles at that time. Then we have $\sum_{j \in X_i} q_j \leq k \cdot (1 - \epsilon) + \sum_{j' < j} 2^{j'-n} \cdot \epsilon < k \cdot (1 - \epsilon) + 2^{j-n} \cdot \epsilon$, and $\sum_{j \in X_{i+1}} q_j \geq k \cdot (1 - \epsilon) + 2^{j-n} \cdot \epsilon$. Hence, swapping the bundles of agents i and $i + 1$ increases the social welfare, violating local efficiency. This proves the claim.

We show that, following from the claim, agent 1 is assigned at least n goods. Suppose for a contradiction that agent 1 receives at most $n - 1$ goods. Then, by induction, agent 1 receives at most $\max(n - i, 0)$ goods for every i . This implies that the algorithm

Phase	P1						P2	P3		P4
Item	1	2	3	4	5	6	7	8	9	10
Agent 1	①	①	1	1	1	1	1	①	1	①
Agent 2	1	1	①	①	1	1	1	1	①	1
Agent 3	1	1	1	1	①	1	①	1	1	0
Agent 4	1	1	1	1	1	①	1	0	0	0
Agent 5	1	1	1	1	1	1	0	0	0	0

Table 2: Illustration of the lower bound construction for binary additive valuations in the proof of Theorem 4.5. Red entries represent agent–item pairs with value 0, green entries represent pairs with value 1, and circled entries represent the allocation made by the online algorithm.

allocates at most $n(n - 1)/2 < m$ goods, a contradiction. Therefore, agent 1 receives at least n goods.

Furthermore, we can inductively argue that each agent i receives at most $n - i + 1$ goods. Thus agent i ’s envy towards agent 1 is at least $w_i \cdot (i - 1 - n \cdot \epsilon) \geq (1 - n \cdot \epsilon) \cdot (i - 1 - n \cdot \epsilon) \geq i - 2n \cdot \epsilon$. Consequently, the total subsidy is at least $2 + \dots + n - 2n^2 \cdot \epsilon$. \square

4.2.4 Restricted Additive Valuations. An instance with valuations v_1, \dots, v_n is *restricted additive* if there are some $u_1, \dots, u_m \geq 0$ such that for all $i \in [n]$, v_i is additive and $v_i(\{g\}) \in \{0, u_g\}$ for $g \in [m]$. Restricted additive valuations capture scenarios where all agents agree on the relative value of items (represented by u_g), but each agent is only interested in a subset of them; see [e.g. 1]. An important special case of restricted additive valuations is the binary additive case, where $u_g = 1$ for all g ; see [e.g. 29].

THEOREM 4.5. *There exists an online algorithm that achieves envy-freeness with a total subsidy of at most $n(n - 1)/2$ for all instances with restricted additive valuations. Moreover, for any online algorithm, there exists an instance with binary additive valuations for which achieving envy-freeness requires a total subsidy of at least $n(n - 1)/2$.*

PROOF. Upper bound. Consider the following greedy algorithm. Let (X_1, \dots, X_n) denote the current allocation and g be the incoming item. The algorithm assigns g to the agent whose current bundle has the smallest total value among agents who value g positively, that is, to agent $i \in \arg \min_{a: v_a(g) > 0} v_a(X_a)$. Because each item is allocated to an agent who values it positively and no other agent values it higher, the resulting allocation maximizes welfare.

Let $X = (X_1, \dots, X_n)$ be the final allocation. Without loss of generality, we can relabel the agents so that their final bundle values are in nonincreasing order: $v_1(X_1) \geq v_2(X_2) \geq \dots \geq v_n(X_n)$.

We first show that for any pair of agents $i < j$, agent i does not envy agent j . Since agent j only receives items that they value positively, we have $v_j(X_j) \geq v_i(X_j)$. Together with our ordering assumption $v_i(X_i) \geq v_j(X_j)$, it follows that i does not envy j .

Next, consider any pair $j < i$. We show that agent i envies agent j by at most 1. Let g be the last item assigned to j that i values positively, and let Y denote the partial allocation just before g arrives. At that time, the algorithm chose to give g to j rather than to i , so by definition of the greedy rule, $v_i(Y_i) \geq v_j(Y_j)$. Because agent j values all items in X_j positively, we also have $v_j(Y_j) \geq$

$v_i(Y_j)$. Combining these inequalities gives $v_i(Y_i) \geq v_i(Y_j)$. When g is added to j 's bundle, agent i 's value for g increases $v_i(Y_j)$ by $v_i(\{g\}) \leq 1$, hence $v_i(X_i) \geq v_i(Y_i) \geq v_i(Y_j) + v_i(\{g\}) - 1 = v_i(X_j) - 1$. Therefore, agent i 's envy toward agent j is at most 1.

By the characterization of minimum subsidies (Lemma 2.5), the subsidy required for agent i is at most $i - 1$. Summing over all agents gives a total subsidy of $1 + 2 + \dots + (n - 1) = n(n - 1)/2$.

Lower bound. Fix an online algorithm for the class of restricted additive valuations. We adversarially construct an instance with binary additive valuations where this algorithm either requires at least $\Omega(n^2)$ subsidy or violates envy-freeness. The adversary constructs the instance adaptively, based on the algorithm's decisions, over a sequence of $n - 1$ phases.

Throughout the execution, the adversary maintains a set of eliminated agents and eliminates a new agent at the end of each phase. The agent eliminated at the end of phase p is denoted by e_p .

At the beginning of phase p , for $1 \leq p \leq n - 1$, define $C_p = \arg \min_{i \in [n] \setminus \{e_1, \dots, e_{p-1}\}} v_i(X_i)$, i.e., C_p is the set of non-eliminated agents with minimum value. We let C_p be the set of candidates for elimination. The adversary begins adding identical items, each of value 1 for the non-eliminated agents in $[n] \setminus \{e_1, \dots, e_{p-1}\}$ and value 0 for the eliminated agents in $\{e_1, \dots, e_{p-1}\}$. The adversary continues adding such items until one of the following occurs:

- (1) the algorithm assigns an item to an eliminated agent,
- (2) n^3 items have been added during phase p , or
- (3) all but exactly one candidate agent in C_p have been assigned an item during phase p .

At the end of phase p , the adversary selects a candidate agent $e_p \in C_p$ who received no item during phase p to be eliminated.

Case 1. The adversary terminates because an item is assigned to an eliminated agent. We show that in this case the algorithm violates local efficiency. Suppose the algorithm assigns an item to agent e_q with $q < p$ during phase p . Let X be the final allocation. Since every item except the last one was assigned to a non-eliminated agent at the time it arrived, the social welfare of X is $SW(X) = m - 1$.

Pick a candidate e_p that received no item in this phase. Consider an allocation X' obtained from X by reassigning $X'_{e_p} = X_{e_q}$ and $X'_{e_{p-j-1}} = X_{e_{p-j}}$ for $0 \leq j < p - q$. Note that agent e_p values every item in X_{e_q} at 1, since e_p was still active at the beginning of phase p . Similarly, for any $0 \leq j < p - q$, agent e_{p-j-1} values all items in $X_{e_{p-j}}$ at 1, because e_{p-j} received no items during phase $p - j$ by the choice of elimination. Thus, $SW(X') = m$, implying that X is not locally efficient, and hence not envy-freeable.

Case 2. The adversary terminates after n^3 items have arrived. We show that the algorithm uses a subsidy of $\Omega(n^2)$. By construction of valuations and the definition of C_p , at the beginning of phase p , each $i \in C_p$ weakly envies all other non-eliminated agents. Since n^3 items arrive, there exists some non-eliminated agent $j \notin \{e_1, \dots, e_{p-1}\}$ who receives at least n^2 items during phase p , and some candidate agent $i \in C_p$ who receives no item during that phase (otherwise we would be in case (3)).

Therefore, in the final allocation X , we have $v_i(X_i) + n^2 \leq v_i(X_j)$, implying that envy-freeness requires a subsidy of at least n^2 .

Case 3. The adversary proceeds to the next phase. We show that at the end of phase $n - 1$, the algorithm still requires a total subsidy of at least $\Omega(n^2)$. By construction, for every $1 \leq p \leq n - 2$, the

agent e_p eliminated in phase p envies the agent e_{p+1} eliminated in the next phase by at least 1. By the characterization of minimum subsidy (Lemma 2.5), this implies that the total subsidy is at least $1 + 2 + \dots + (n - 1) = n(n - 1)/2$. \square

4.2.5 Identical Monotone Valuations. An instance with valuations v_1, \dots, v_n is *identical monotone* if, for all $i, j \in [n]$, $v_i(S) = v_j(S)$ for every bundle S . This class is quite general, imposing only monotonicity and normalization on the valuation, but it assumes that all agents have identical preferences; see [e.g. 43].

THEOREM 4.6. *There exists an online algorithm that achieves envy-freeness with a total subsidy of at most $n - 1$ for all instances with identical monotone valuations. Moreover, for any online algorithm, there exists an instance with identical monotone valuations for which achieving envy-freeness requires a total subsidy of at least $n - 1$.*

The proof of the upper bound builds on a construction of an EF1 allocation for identical monotone valuations.

5 DISCUSSION

Comparison with Envy Minimization. We note that the envy minimization problem without the local efficiency requirement has been studied extensively (see [e.g. 2, 30]), and seems to have connections to subsidy minimization (i.e., Lemma 2.5). Benadè, Kazachkov, Proccaccia, Psomas, and Zeng [12] obtain an envy bound against an adaptive adversary of $\tilde{O}(\sqrt{m/n})$ for additive valuations over sufficiently many items, and show that this bound is essentially optimal. Lemma 2.5 seems to suggest that a bound on envy should imply a bound on subsidy, however, this holds only when the algorithm also maintains local efficiency at every step. This latter requirement poses the main challenge in online subsidy minimization.

Online Fair Division of Chores. We remark that most of our results extend to the fair division of chores, for which the valuation functions are *non-increasing*, and for which local efficiency continues to characterize envy-freeability. The subsidy, however, is measured differently: agents assigned extra chores receive greater payments. Consequently, while some of our subsidy bounds change with respect to their dependence on n , the boundaries we established for our two central questions remain unchanged.

Open Problems. Our work creates some compelling directions for future research. Can the results we obtain for Questions 1 and 2 be extended to any other interesting valuation classes? Another problem we leave open is determining the tight bound on the minimum subsidy for k -valued instances.

ACKNOWLEDGMENTS

A portion of this work was completed while P. Kulkarni was a student at the University of Illinois Urbana-Champaign. P. Kulkarni, R. Mehta and V. V. Narayan were supported by NSF grant CCF-2334461. P. Kulkarni was supported by a gift from Adobe to S. Khuller. T. Ponitka was supported by ERC grant 101170373, an Amazon Research Award, NSF-BSF grant 2020788, Israel Science Foundation grant 2600/24, and a TAU Center for AI and Data Science grant.

REFERENCES

- [1] Hannaneh Akrami, Rojin Rezvan, and Masoud Seddighin. 2022. An EF2X Allocation Protocol for Restricted Additive Valuations. In *IJCAI*. 413–419.
- [2] Martin Aleksandrov, Haris Aziz, Serge Gaspers, and Toby Walsh. 2015. Online Fair Division: Analysing a Food Bank Problem. In *IJCAI*. AAAI Press, 2540–2546.
- [3] Georgios Amanatidis, Haris Aziz, Georgios Birmpas, Aris Filos-Ratsikas, Bo Li, Hervé Moulin, Alexandros A Voudouris, and Xiaowei Wu. 2023. Fair division of indivisible goods: Recent progress and open questions. *Artificial Intelligence* 322 (2023), 103965.
- [4] Georgios Amanatidis, Georgios Birmpas, Aris Filos-Ratsikas, Alexandros HOLLender, and Alexandros A. Voudouris. 2021. Maximum Nash Welfare and Other Stories About EFX. *Theoretical Computer Science* 863 (2021), 69–85.
- [5] Nima Anari, Tung Mai, Shayan Oveis Gharan, and Vijay V. Vazirani. 2018. Nash Social Welfare for Indivisible Items under Separable, Piecewise-Linear Concave Utilities. In *SODA*. SIAM, 2274–2290.
- [6] Enriqueta Aragones. 1995. A derivation of the money Rawlsian solution. *Social Choice and Welfare* 12, 3 (1995), 267–276.
- [7] Haris Aziz, Xin Huang, Kei Kimura, Indrajit Saha, Zhaohong Sun, Mashbat Suzuki, and Makoto Yokoo. 2025. Weighted Envy-free Allocation with Subsidy. In *AAMAS*. International Foundation for Autonomous Agents and Multiagent Systems / ACM, 2417–2419.
- [8] Moshe Babaioff, Tomer Ezra, and Uriel Feige. 2021. Fair and Truthful Mechanisms for Dichotomous Valuations. In *AAAI*. AAAI Press, 5119–5126.
- [9] Siddharth Barman, Anand Krishna, Yadati Narahari, and Soumyarup Sadhukhan. 2022. Achieving Envy-Freeness with Limited Subsidies under Dichotomous Valuations. In *IJCAI*. ijcai.org, 60–66.
- [10] Siddharth Barman, Vishnu V. Narayan, and Paritosh Verma. 2023. Fair Chore Division under Binary Supermodular Costs. In *AAMAS*. ACM, 2863–2865.
- [11] Siddharth Barman and Paritosh Verma. 2021. Existence and Computation of Maximin Fair Allocations Under Matroid-Rank Valuations. In *AAMAS*. ACM, 169–177.
- [12] Gerdus Benadé, Aleksandr M. Kazachkov, Ariel D. Procaccia, Christos-Alexandros Psomas, and David Zeng. 2024. Fair and Efficient Online Allocations. *Oper. Res.* 72, 4 (2024), 1438–1452.
- [13] Johannes Brustle, Jack Dippel, Vishnu V. Narayan, Mashbat Suzuki, and Adrian Vetta. 2020. One Dollar Each Eliminates Envy. In *EC*. ACM, 23–39.
- [14] David Buchfuhrer, Shaddin Dughmi, Hu Fu, Robert Kleinberg, Elchanan Mossel, Christos H. Papadimitriou, Michael Schapira, Yaron Singer, and Christopher Umans. 2010. Inapproximability for VCG-Based Combinatorial Auctions. In *SODA*. SIAM, 518–536.
- [15] Jaroslaw Byrka, Franciszek Malinka, and Tomasz Ponitka. 2025. Probing EFX via PMMS: (Non-)Existence Results in Discrete Fair Division. *CoRR* abs/2507.14957 (2025).
- [16] Ioannis Caragiannis and Stavros Ioannidis. 2021. Computing Envy-Freeable Allocations with Limited Subsidies. In *Web and Internet Economics - 17th International Conference, WINE 2021*, Vol. 13112. 522–539.
- [17] Bhaskar Ray Chaudhury, Yun Kuen Cheung, Jugal Garg, Naveen Garg, Martin Hoefer, and Kurt Mehlhorn. 2022. Fair division of indivisible goods for a class of concave valuations. *Journal of Artificial Intelligence Research* 74 (2022), 111–142.
- [18] Chandra Chekuri, Pooja Kulkarni, Rucha Kulkarni, and Ruta Mehta. 2024. 1/2-Approximate MMS Allocation for Separable Piecewise Linear Concave Valuations. In *AAAI*. AAAI Press, 9590–9597.
- [19] Edith Cohen, Michal Feldman, Amos Fiat, Haim Kaplan, and Svetlana Olonetsky. 2012. Envy-Free Makespan Approximation. *SIAM J. Comput.* 41, 1 (2012), 12–25.
- [20] Aris Deligkas, Themistoklis Melissourgios, and Paul G. Spirakis. 2021. Walrasian Equilibria in Markets with Small Demands. In *AAMAS*. ACM, 413–419.
- [21] Francis Edward Su. 1999. Rental harmony: Sperner’s lemma in fair division. *The American mathematical monthly* 106, 10 (1999), 930–942.
- [22] Michal Feldman, Jugal Garg, Vishnu V. Narayan, and Tomasz Ponitka. 2025. Proportionally Fair Makespan Approximation. In *AAAI*. AAAI Press, 13839–13846.
- [23] Duncan Foley. 1967. Resource Allocation and the Public Sector. *Yale Economic Essays* 7, 1 (1967), 45–98.
- [24] Ya’akov Gal, Moshe Mash, Ariel D Procaccia, and Yair Zick. 2017. Which is the fairest (rent division) of them all? *Journal of the ACM (JACM)* 64, 6 (2017), 1–22.
- [25] Jugal Garg, Martin Hoefer, and Kurt Mehlhorn. 2018. Approximating the Nash Social Welfare with Budget-Additive Valuations. In *SODA*. SIAM, 2326–2340.
- [26] Jugal Garg and Aniket Murhekar. 2023. Computing Fair and Efficient Allocations with Few Utility Values. *Theoretical Computer Science* 962 (2023), 113932.
- [27] Vasilis Gkatzelis, Alexandros Psomas, and Xizhi Tan. 2021. Fair and Efficient Online Allocations with Normalized Valuations. In *AAAI*. AAAI Press, 5440–5447.
- [28] Claus-Jochen Haake, Matthias G. Raith, and Francis Edward Su. 2002. Bidding for envy-freeness: A procedural approach to n-player fair-division problems. *Soc. Choice Welf.* 19, 4 (2002), 723–749.
- [29] Daniel Halpern, Ariel D. Procaccia, Alexandros Psomas, and Nisarg Shah. 2020. Fair Division with Binary Valuations: One Rule to Rule Them All. In *WINE*. Springer, 370–383.
- [30] Daniel Halpern, Alexandros Psomas, Paritosh Verma, and Daniel Xie. 2025. Online Envy Minimization and Multicolor Discrepancy: Equivalences and Separations. In *EC*. ACM, 188.
- [31] Daniel Halpern and Nisarg Shah. 2019. Fair Division with Subsidy. In *SAGT (Lecture Notes in Computer Science, Vol. 11801)*. Springer, 374–389.
- [32] Jason Hartline, Sam Leong, Ahuva Mualem, Michael Schapira, and Aviv Zohar. 2008. *Multi-dimensional Envy-free Scheduling Mechanisms*. Technical Report 1144. The Hebrew University.
- [33] Jiafan He, Ariel D. Procaccia, Alexandros Psomas, and David Zeng. 2019. Achieving a Fairer Future by Changing the Past. In *IJCAI*. ijcai.org, 343–349.
- [34] Ian Kash, Ariel D Procaccia, and Nisarg Shah. 2014. No agent left behind: Dynamic fair division of multiple resources. *Journal of Artificial Intelligence Research* 51 (2014), 579–603.
- [35] Flip Klijn. 2000. An algorithm for envy-free allocations in an economy with indivisible objects and money. *Social Choice and Welfare* 17, 2 (2000), 201–215.
- [36] Pooja Kulkarni, Ruta Mehta, Vishnu V Narayan, and Tomasz Ponitka. 2025. Online Fair Division With Subsidy: When Do Envy-Free Allocations Exist, and at What Cost? *arXiv preprint arXiv:2510.13633* (2025).
- [37] Pooja Kulkarni, Ruta Mehta, and Parnian Shahkar. 2025. Online Fair Division: Towards Ex-Post Constant MMS Guarantees. In *EC*. ACM, 638.
- [38] Max Dupre la Tour and Kaito Fujii. 2025. Discrepancy And Fair Division For Non-Additive Valuations. *CoRR* abs/2509.16802 (2025).
- [39] Sébastien Lahaie, Daniel M. Pennock, Amin Saberi, and Rakesh V. Vohra. 2007. Sponsored Search Auctions. In *Algorithmic Game Theory*, Noam Nisan, Tim Roughgarden, Éva Tardos, and Vijay V. Vazirani (Eds.). Cambridge University Press, 699–716.
- [40] Renato Paes Leme. 2017. Gross substitutability: An algorithmic survey. *Games Econ. Behav.* 106 (2017), 294–316.
- [41] Eric S. Maskin. 1987. On the Fair Allocation of Indivisible Goods. In *Arrow and the Foundations of the Theory of Economic Policy*. 341–349.
- [42] Aranyak Mehta, Amin Saberi, Umesh Vazirani, and Vijay Vazirani. 2007. Adwords and generalized online matching. *Journal of the ACM (JACM)* 54, 5 (2007), 22–es.
- [43] Benjamin Plaut and Tim Roughgarden. 2020. Almost Envy-Freeness with General Valuations. *SIAM J. Discret. Math.* 34, 2 (2020), 1039–1068.
- [44] Hugo Steinhaus. 1948. The problem of fair division. *Econometrica* 16 (1948), 101–104.
- [45] Walter Stromquist. 1980. How to Cut a Cake Fairly. *The American Mathematical Monthly* 87, 8 (1980), 640–644.
- [46] Koichi Tadenuma and William Thomson. 1993. The fair allocation of an indivisible good when monetary compensations are possible. *Mathematical Social Sciences* 25, 2 (1993), 117–132.
- [47] Nicholas Teh. 2024. Envy-free house allocation with minimum subsidy. *Operations Research Letters* 54 (2024).
- [48] Toby Walsh. 2011. Online Cake Cutting. In *ADT (Lecture Notes in Computer Science, Vol. 6992)*. Springer, 292–305.
- [49] Wikipedia contributors. 2025. *Rearrangement inequality*. https://en.wikipedia.org/wiki/Rearrangement_inequality Accessed: 2025-10-08.
- [50] Hanrui Zhang and Vincent Conitzer. 2020. Learning the Valuations of a k-demand Agent. In *ICML (Proceedings of Machine Learning Research, Vol. 119)*. PMLR, 11066–11075.