Online Fair Allocations with Binary Valuations and Beyond

Yuanyuan Wang[1*†], **Tianze Wei** [2†‡]

[1] IOTCS, University of Macau
[2] Department of Computer Science, City University of Hong Kong
yuanywang@um.edu.mo, t.z.wei-8@my.cityu.edu.hk

Abstract

In an online fair allocation problem, a sequence of indivisible items arrives online and needs to be allocated to offline agents immediately and irrevocably. In our paper, we study the online allocation of either goods or chores. We employ popular fairness notions, including envy-freeness up to one item (EF1) and maximin share fairness (MMS) to capture fairness, and utilitarian social welfare (USW) to measure efficiency. For both settings of items, we present a series of positive results regarding the existence of fair and efficient allocations with widely studied classes of additive binary and personalized bi-valued valuation/cost functions. Furthermore, we complement our results by constructing counterexamples to establish our results as among the best guarantees possible.

Extended version — https://arxiv.org/abs/2505.24321

Introduction

The fair division of indivisible items is a prominent topic in algorithmic game theory and artificial intelligence, with practical applications (Budish 2011; Goldman and Procaccia 2015). Most prior work has focused on *offline* settings, where the information of all items is known a priori. In practice, items are often online, and waiting for all resources to appear can significantly impede progress. We must proactively allocate resources in a timely manner for better outcomes. A paradigmatic example is the food banks problem (Lee et al. 2019), where food arrives online without prior knowledge of future items. Since the food is perishable, it must be allocated upon arrival to some agent. Motivated by these real-world scenarios, we define and discuss the following problem:

(*Online Fair Division*) There are n agents that are fixed and known in advance. The items arrive online (the number of items is *unknown*). Upon the arrival of an item, its values to all agents are revealed, and we must allocate it to some agent or discard it immediately and irrevocably. At all times, the allocation among agents must be fair.

[*]Part of this work was done while the author was a PhD student at Ocean University of China.

[†]These authors contributed equally.

[‡]Corresponding author.

If the online algorithm knows additional information about all future items beforehand, it somewhat reduces the uncertainty of the online model, allowing for more effective decision-making and improved outcomes. (Zhou, Bai, and Wu 2023; Benadè et al. 2024) study the normalized valuations, i.e., online algorithms know the total value of future items. (Benadè, Halpern, and Psomas 2022) studies the online fair allocation, foreseeing the rank of a new arrival item with respect to the past items under different assumptions of agents' valuation distributions. And there are other studies about additional information (He et al. 2019; Elkind et al. 2025; Cookson, Ebadian, and Shah 2025; Neoh, Peters, and Teh 2025; Melissourgos and Protopapas 2025; Choo et al. 2025). For an independent interest, we also examine the online setting with additional information, with relevant results presented in the extended version.

In numerous practical applications, such as vaccine allocation (Rey, Hammad, and Saberi 2023), the online algorithm operates without prior knowledge of future items. (Aleksandrov et al. 2015) studies the online fair division with additive binary valuations and shows that an envy-free up to one item (EF1) allocation exists. However, there are a few non-trivial approximation fairness results of online goods allocation beyond additive binary valuations and online chores allocation. In our paper, we aim to address this gap and study the online allocation model without any information about future items. We discuss the goods and chores settings, respectively. Our primary research question is:

Can we design deterministic online algorithms that are fair and efficient, for valuation functions beyond additive binary, in the online allocation problem?

Our Contributions

We study two settings: goods and chores, in the online deterministic fair indivisible items allocation model.

In the goods setting, for different classes of valuations, including binary and additive personalized bi-valued, our main results are summarized in Table 1. We introduce the non-wastefulness constraint in this setting. With this constraint, first, we show the inapproximability of EF1 and MMS with general additive valuations, even for only two agents with additive personalized tri-valued valuations. Then, we study some restricted valuation functions, such as submodular binary valuation functions. More specifically,

Valuations	Number of agents	Fairness/ Efficiency	LB	UB
Additive (personalized tri-valued)	n	EF1	$\times$ (Thm 1)	
		MMS	$\times$ (Thm 1)	
Submodular binary	n	EF1	$\frac{1}{2}$ (Thm 2)	$\frac{1}{2}$ (Thm 3)
		MMS	$\frac{1}{2}$ (Thm 2)	$\frac{1}{2}$ (Thm 3)
		USW	$\frac{1}{2}$ (Thm 2)	$\frac{1}{2}$ (Thm 3)
Additive personalized bi-valued	2	EF1	$\frac{1}{2}$ (Thm 4)	$\frac{1}{2}$ (Thm 5)
		MMS	$\frac{1}{3}$ (Thm 4)	$\frac{1}{3}$ (Thm 5)
		USW	$\times$ (Prop 1)	
Identical additive binary + personalized bi-valued	n	EF1	1 (Thm 6)	
		MMS	1 (Thm 6)	
		USW	$\times$ (Prop 2)	

Table 1: Goods setting. LB: lower bound. UB: upper bound. "$\times$" indicates inapproximability.

- For submodular binary valuation functions, we propose the Marginal-Greedy Algorithm (Algorithm 1) to compute an allocation that simultaneously satisfies non-wastefulness, $\frac{1}{2}$-EF1, $\frac{1}{2}$-MMS, and $\frac{1}{2}$-max-USW, and we show that these approximation ratios are all tight.
- For two agents with additive personalized bi-valued valuation functions, we design the Adapted Envy-Graph Procedure (Algorithm 2) to compute a non-wasteful allocation satisfying $\frac{1}{2}$-EF1 and $\frac{1}{3}$-MMS simultaneously. Next, we complement this positive result by showing that no deterministic algorithm can guarantee a non-wasteful allocation that is α-EF1 or β-MMS for any $\alpha > \frac{1}{2}$ and $\beta > \frac{1}{3}$, and no deterministic algorithm can compute an non-wasteful allocation where α-EF1 (or α-MMS) is compatible with β-max-USW for any $\alpha, \beta > 0$.
- When one agent has the additive personalized bi-valued valuation function and the remaining agents have identical additive binary valuation functions, we design the Adapted-Picking Algorithm (Algorithm 3) to compute an allocation satisfying non-wastefulness, EF1, and MMS. Further, we show that no deterministic algorithm can compute a non-wasteful allocation that is EF1 (or MMS) and α-max-USW for any $\alpha > 0$.

In the chores setting, we introduce the completeness constraint. With this constraint, we first show the inapproximability of EF1 even for two agents with additive personalized tri-valued cost functions, which blocks our way of studying more general cost functions. Next, we investigate the restricted cost functions, including additive binary cost functions, supermodular binary cost functions, and additive personalized bi-valued cost functions. The contribution and results details are deferred to the extended version.

Related Work

Online Deterministic Fair Allocation. In the majority of the literature, there are two kinds of online arrivals: The former kind is that agents arrive over time (Kash, Procaccia, and Shah 2014; Friedman, Psomas, and Vardi 2015, 2017; Li, Li, and Li 2018), and the latter is that items arrive over time, as seen in (Aleksandrov et al. 2015), which is also the topic of our paper. Our work is mainly related to the study of (Aleksandrov et al. 2015; Hosseini et al. 2024; Amanatidis et al. 2025; Song et al. 2025), where items arrive one by one without any future information known in advance. (Aleksandrov et al. 2015) consider the food bank problem when all agents have additive binary valuation functions and design two simple mechanisms to guarantee ex-ante envy-freeness. (Hosseini et al. 2024) study the class fairness in online matching for indivisible and divisible items, and present a series of positive results. The model of indivisible items in their paper can be seen as a special case of our model, which is discussed later in this paper. In other words, we study a more general setting, beyond additive binary valuation functions, and derive the approximation results that strengthen their results. (Amanatidis et al. 2025) study the restricted valuation function in the goods setting; their main positive results are built on the additive bi-valued valuation functions, while our positive results are with submodular binary valuation functions and additive personalized bi-valued valuation functions. (Song et al. 2025) consider the chores setting and presents the parametrized approximation of MMS fairness. In our paper, we consider MMS along with EF1 and the efficiency notion, which differs from that in their paper.

Offline Fair Allocation. When agents have additive valuation functions, there is a rich body of literature that discusses EF1 or MMS. Please refer to the survey by (Amanatidis et al. 2023) for an overview. Here, we highlight some papers most relevant to our work on offline fair division with submodular binary valuation functions or supermodular binary cost functions. When agents have submodular binary valuations in the goods setting, (Benabbou et al. 2021) study the compatibility of fairness and efficiency and show that a maximum utilitarian social welfare allocation that is EF1 can be efficiently computed, and (Barman and Verma 2021) consider the fairness notion - MMS and prove that a maximum utilitarian social welfare allocation that satisfies MMS can be computed in polynomial time. When agents have supermodular binary cost functions in the chores setting, (Barman, Narayan, and Verma 2023) show that an EF1 or MMS allocation minimizing the social cost can be computed in polynomial time.

Preliminaries

For each natural number $s \in \mathbb{N}$, let $[s] = \{1, \ldots, s\}$. Let $T = \{e_1, \ldots, e_t, \ldots\}$ be the set of *indivisible* items arriving one by one, where $|T|$ is unknown, and $N = \{1, \ldots, n\}$ be the set of offline agents. When an online item e_t arrives, its values or costs to all agents are revealed. When items are goods, each agent is endowed with a valuation function $v_i : 2^T \rightarrow \mathbb{R}_{\geq 0}$. When items are chores, each agent is endowed with a cost function $c_i : 2^T \rightarrow \mathbb{R}_{\geq 0}$. We assume that $v_i(\emptyset) = 0$ ($c_i(\emptyset) = 0$), and monotone, $v_i(X) \leq v_i(Y)$ ($c_i(X) \leq c_i(Y)$) for all $X \subseteq Y \subseteq T$. For any item $e \in T$ and any subset $X \subseteq T$, we denote by $\Delta^i_X(e) = v_i(X \cup \{e\}) - v_i(X)$ or $c_i(X \cup \{e\}) - c_i(X)$ the marginal value or cost of an item $e \in T$ in the set X. v_i is *ad-*

	e_1	e_2	e_3	e_4
agent 1	**6**	8	10	6
agent 2	13	**8**	**5**	0

Table 2: Example of the deterministic online allocation.

ditive if the value of any set of items S is the sum of the value of each item in S, that is, $v_i(X) = \sum_{e \in X} v_i(e)$. v_i is *submodular* if $v_i(X)+v_i(Y) \geq v_i(X \cup Y)+v_i(X \cap Y)$ for any two sets $X, Y \subseteq T$. v_i is *supermodular* if $v_i(X) + v_i(Y) \leq v_i(X \cup Y) + v_i(X \cap Y)$ for any two sets $X, Y \subseteq T$. There are similar arguments for c_i, and we omit it. Let T^k be the set of items that have arrived by round $k \in [t]$. We denote an allocation by $\mathbf{A}^k = (A_1^k, \ldots, A_n^k)$ in round k, where $A_i^k \subseteq T^k$ is the bundle of items that are allocated to agent i, and $A_i^k \cap A_j^k = \emptyset$ for any $i, j \in N$. Let $\Pi_n(T^k)$ be the set of all allocations of T^k. Collectively, $(T, N, (v_i)_{i \in N})$ forms an instance of *online goods fair allocation* and $(T, N, (c_i)_{i \in N})$ forms an instance of *online chores fair allocation*. For a singleton set $\{e\}$, we will use $v_i(e)$ $(c_i(e))$ as a shorthand for $v_i(\{e\})$ $(c_i(\{e\}))$.

Deterministic online algorithms. In the online setting, the items in T arrive one by one in an arbitrary order, where $|T|$ is unknown. When an item arrives, the values (costs) of this item for all agents are revealed. The online algorithm must make an immediate and irrevocable decision to allocate the item to one agent or discard it *deterministically*.

Our objective is to design deterministic algorithms to compute allocations guaranteeing the desired approximation fairness and efficiency, including EF1, MMS, and USW (Definitions 1, 2, and 3), as well as constraint, i.e., non-wastefulness (Definition 4), at the end of each round. In contrast, our impossibility results will hold even if the desired guarantees are required to hold only at the end. For a better understanding of our online model, let us demonstrate the deterministic online allocation procedure through the following example.

Example. *Consider that there are two agents with additive valuation functions, and the arriving items are goods. The value of each item is shown in Table 2, and note that the value of each item is only revealed after it arrives. We utilize a deterministic online algorithm referred to as* OALG. *In round* $t = 1$, OALG *allocates* e_1 *to agent 1, and the allocation* $\mathbf{A}^1$ *is EF1, MMS, and* $\frac{6}{13}$*-max-USW. In round* $t = 2$, OALG *allocates* e_2 *to agent 2, and the allocation* $\mathbf{A}^2$ *is EF1, MMS, and* $\frac{2}{3}$*-max-USW. In round* $t = 3$, OALG *allocate* e_3 *to agent 2, and the allocation* $\mathbf{A}^3$ *is* $\frac{3}{4}$*-EF1,* $\frac{3}{5}$*-MMS, and* $\frac{19}{31}$*-max-USW. In round* $t = 4$, OALG *discards* e_4*, and the allocation* $\mathbf{A}^4$ *is* $\frac{3}{4}$*-EF1,* $\frac{3}{7}$*-MMS, and* $\frac{19}{37}$*-max-USW. Then, no further item arrives and we say that* OALG *achieves* $\min\{\frac{3}{4}, 1\} = \frac{3}{4}$*-EF1,* $\min\{\frac{3}{7}, \frac{3}{5}, 1\} = \frac{3}{7}$*-MMS, and* $\min\{\frac{6}{13}, \frac{19}{37}, \frac{19}{31}, \frac{2}{3}\} = \frac{6}{13}$*-max-USW.*

Allocation of Goods

Throughout this section, we focus on the goods setting. We first introduce some definitions of fairness and efficiency notions that are used in this section.

Definition 1 (Envy-freeness up to One Good)**.** *For any* $\alpha \in [0, 1]$*, an allocation* $\mathbf{A}^k$ *is* α*-approximate envy-free up to one good (*α*-EF1) if, for every pair of agents* $i, i' \in N$*, either* $A_{i'}^k = \emptyset$ *or* $v_i(A_i^k) \geq \alpha \cdot v_i(A_{i'}^k \setminus \{e\})$ *for some* $e \in A_{i'}^k$*.*

Definition 2 (Maximin Share Fairness)**.** *For any agent* $i \in N$*, her maximin share* MMS_i^k *by round* k *is defined as:*

$$\mathsf{MMS}_i^k = \max_{B^k \in \Pi_n(T^k)} \min_{i' \in N} v_i(B_{i'}^k),$$

For any $\alpha \in [0, 1]$*, an allocation* $\mathbf{A}^k$ *is* α*-approximate maximin share fair (*α*-MMS) if for any agent* $i \in N$*, it holds that* $v_i(A_i^k) \geq \alpha \cdot \mathsf{MMS}_i^k$*.*

Definition 3 (Utilitarian Social Welfare)**.** *The utilitarian social welfare of allocation* $\mathbf{A}^k$ *is given by* $USW(\mathbf{A}^k) = \sum_{i \in N} v_i(A_i^k)$*. For any* $\alpha \in [0, 1]$*, an allocation* $\mathbf{A}^k$ *is* α*-max-USW if* $USW(\mathbf{A}^k) \geq \alpha \cdot \max_{\mathbf{B}^k \in \Pi_n(T^k)} USW(\mathbf{B}^k)$*.*

Besides the above definitions, we also pay attention to the following restriction.

Definition 4 (Non-Wastefulness)**.** *An allocation* $\mathbf{A}^k$ *is non-wasteful (NW) if (1)* $\Delta^i_{A_i^k \setminus \{e\}}(e) > 0$ *for any agent* $i \in N$ *and any good* $e \in A_i^k$*, and (2)* $\Delta^i_{A_i^k}(e') = 0$ *for any agent* $i \in N$ *and any discarded good* $e' \in T^k$*.*

Note that imposing non-wastefulness in the goods setting is necessary, as our online model permits item discarding (see our example for illustration). Without this constraint, achieving EF1 would be trivial by simply discarding all online items. For agents with additive valuation functions, an NW allocation can always be found as long as each arriving item is allocated to at least one agent who values it positively. Below, we present a positive result where an NW allocation always exists for submodular binary valuation functions, and then complement it with a negative result for general submodular valuation functions.

Lemma 1. *For submodular binary valuation functions, an NW allocation always exists.*

Lemma 2. *For submodular valuation functions, there exists an instance that no deterministic online algorithm can compute an NW allocation, even for two agents with identical valuation functions.*

The Impossibilities of Approximate EF1 and MMS

(Aleksandrov et al. 2015) show that in the online setting, an EF1 allocation exists for additive binary valuations, where for each agent $i \in N$, v_i is additive and $v_i(e) \in \{0, 1\}$ for any $e \in T$. In this part, we present a strong negative result regarding the approximations of EF1 and MMS for general additive valuations. We say that the valuation function v_i is additive personalized tri-valued if v_i is additive and $v_i(e) \in \{a_i, b_i, z_i\}$ for any $e \in T$, where $0 < a_i \leq b_i \leq z_i$.

	e_1	e_2	e_3
agent 1	$\mathbf{1}$	$\frac{1}{\epsilon}$	$\frac{1}{\epsilon}$
agent 2	1	$\boldsymbol{\epsilon}$	$\frac{1}{\epsilon}$

Table 3: The impossibility result of EF1 and MMS.

Theorem 1. *No deterministic online algorithm can compute an NW allocation that guarantees α-EF1 or α-MMS for any $\alpha > 0$, even for two agents with additive personalized tri-valued valuation functions.*

Proof. Given any $\alpha > 0$, let ϵ be a sufficiently small number that is arbitrarily close to 0, such that $0 < \epsilon < \alpha$. Consider an online instance with two agents, where for any agent $i \in N$ and item $e \in T$, we have $v_i(e) \in \{\epsilon, 1, \frac{1}{\epsilon}\}$, where $0 < \epsilon < 1$. The items arrive online, and the value of each item is shown in Table 3. When $t = 1$, the item e_1 arrives, i.e., $T^1 = \{e_1\}$. Without loss of generality, suppose that it is allocated to agent 1, i.e., $A_1^1 = \{e_1\}$ and $A_2^1 = \emptyset$. When $t = 2$, the item e_2 arrives, i.e., $T^2 = \{e_1, e_2\}$. To maintain the fairness of allocation, we must allocate it to agent 2, i.e., $A_1^2 = \{e_1\}$ and $A_2^2 = \{e_2\}$. Otherwise, agent 1 gets $\{e_1, e_2\}$ and agent 2 gets an empty set, we have $v_2(\emptyset) = 0 < v_2(\{e_1, e_2\} \setminus \{e\})$ for any $e \in \{e_1, e_2\}$ and $\mathsf{MMS}_2(T^2) = \epsilon$. Then, this allocation is 0-EF1 and 0-MMS. When $t = 3$, item e_3 arrives, we have $T^3 = \{e_1, e_2, e_3\}$, where $\mathsf{MMS}_1(T^3) = \frac{1}{\epsilon}$ and $\mathsf{MMS}_2(T^3) = 1 + \epsilon$. If we allocate it to agent 1, i.e., $A_1^3 = \{e_1, e_3\}$ and $A_2^3 = \{e_2\}$, we have $v_2(A_2^3) = \epsilon, v_2(A_1^3 \setminus \{e_1\}) = \frac{1}{\epsilon}$, and $v_2(A_1^3 \setminus \{e_3\}) = 1$. Thus, $\mathbf{A^3}$ is ϵ-EF1 and $\frac{\epsilon}{1+\epsilon}$-MMS. If item e_3 is allocated to agent 2, i.e., $A_1^3 = \{e_1\}$ and $A_2^3 = \{e_2, e_3\}$, we have $v_1(A_1^3) = 1 = \epsilon \cdot v_1(A_2^3 \setminus \{e\})$ for any $e \in A_2^3$. Thus, $\mathbf{A^3}$ is ϵ-EF1 and ϵ-MMS.

Therefore, in the above instance, no deterministic algorithm can guarantee an NW allocation that is α-EF1 or α-MMS. This result can be generalized to more than two agents, and we defer the proof to the extended version. □

By Theorem 1, we cannot make further progress in the approximations of EF1 and MMS for additive personalized tri-valued valuation functions and beyond in the online setting. In the following parts, we focus on two classes of valuation functions that are more general than additive binary functions, including submodular binary and additive personalized bi-valued valuation functions, and study whether there are positive results about the approximations of EF1 and MMS with some efficiency guarantees.

Submodular Binary Valuations

This part focuses on submodular binary valuation functions, where for each agent $i \in N$, v_i is submodular, and the marginal value of any item is 0 or 1. For a subclass of submodular binary valuation functions based on maximum (unweighted) bipartite matching, also known as assignment (Munkres 1957) or OXS (Lehmann, Lehmann, and Nisan 2001) valuations, (Hosseini et al. 2024) show that online allocations satisfying $\frac{1}{2}$-EF1 and $\frac{1}{2}$-MMS exist.

Algorithm 1: Marginal-Greedy Algorithm

Input: An instance $(T, N, (v_i)_{i \in N})$ with submodular binary valuation functions
Output: An NW approximate EF1 and approximate MMS allocation $\mathbf{A}$

Initialize: let $\pi = (\pi_1, \dots, \pi_n)$ be an arbitrary order of n agents;
Let $\mathbf{A} = (\emptyset, \dots, \emptyset)$;
when *item* $e \in T$ *arrives* **do**
  for $i = 1$ *to* n **do**
    if $\Delta_{A_{\pi_i}}^{\pi_i}(e) > 0$ **then**
      Let $A_{\pi_i} = A_{\pi_i} \cup \{e\}$;
      $\pi = (\pi_1, \dots, \pi_{i-1}, \pi_{i+1}, \dots, \pi_n, \pi_i)$;
      break;
    end
  end
return $\mathbf{A}$;

Limitation of (Hosseini et al. 2024): Their $\frac{1}{2}$-EF1/$\frac{1}{2}$-MMS guarantees are restricted to OXS valuations, where items must be assigned within predefined groups under rigid matching constraints. Each member in the group can only receive at most one item, and once someone gets an item, it cannot be changed. Thus, their algorithm can only handle the OXS valuation function and its subclasses.

Moving beyond the online allocation model based on matching, we consider the general online allocation problem. Building upon the effective greedy technique in the submodular optimization and the sequential allocating technique to guarantee fairness in (Hosseini et al. 2024), we propose the Marginal-Greedy Algorithm (Algorithm 1), greedily assigning each arriving item to the agent who receives the maximum marginal value, for submodular binary valuation functions, and then show that the Marginal-Greedy Algorithm computes an NW allocation that is $\frac{1}{2}$-EF1, $\frac{1}{2}$-MMS, and $\frac{1}{2}$-max-USW, where these approximations are tight.

In the analysis of results about submodular binary valuation functions, we effectively leverage submodularity and the properties of the matroid. There exists a relationship of submodularity and matroid: a function is submodular binary iff it is a rank function of a matroid (Schrijver et al. 2003).

Definition 5. *A set system $\mathcal{M} = (E, \mathcal{F})$, where E is a ground set and $\mathcal{F}$ is a family of subset of E, is a matroid if it satisfies the following properties: (1) $\emptyset \in \mathcal{F}$; (2) if $D \in \mathcal{F}$ and $C \subseteq D$, then $C \in \mathcal{F}$; (3) if $C, D \in \mathcal{F}$ and $|D| > |C|$, then there exists $x \in D \setminus C$ such that $C + x \in \mathcal{F}$. The rank function of matroid $\mathcal{M}$, denoted by $r(\cdot)$, is defined by: $r : 2^T \to R$, $r(X) = \max\{|Y| : Y \subseteq T, Y \in \mathcal{F}\}$.*

Theorem 2. *For the deterministic allocation of indivisible goods with submodular binary valuation functions, the Marginal-Greedy Algorithm (Algorithm 1) computes an NW allocation that satisfies $\frac{1}{2}$-EF1, $\frac{1}{2}$-MMS, and $\frac{1}{2}$-max-USW.*

Proof. Given an online fair allocation instance $I = (T, N, (v_i)_{i \in N})$, where v_i is submodular and binary for all $i \in N$. Fix an arbitrary round k, let $T^k = \{e_1, \dots, e_k\}$

be the set of items that have already arrived, denoted by $\mathbf{A}^k = (A_1^k, \ldots, A_n^k)$ and A_0^k the allocation and the set of all unallocated items at the end of round k respectively.

In Algorithm 1, the arriving item is allocated to an agent who thinks that the marginal value of this item is 1 if there exists such an agent. By Lemma 1, $\mathbf{A}^k$ is NW. For each v_i, there exists a matroid $\mathcal{M}_i = (T^k, \mathcal{F}_i)$ such that v_i is the rank function of $\mathcal{M}_i$. We have $v_i(A_i^k) = |A_i^k|$, and $A_i^k \in \mathcal{F}_i$.

First, we show that $\mathbf{A}^k$ is $\frac{1}{2}$-EF1. Consider any two agents $i, j \in N$, $i \neq j$. If $v_i(A_j^k) > v_i(A_i^k)$, by the property (3) of the matroid, there must exist an item $e \in A_j^k$ such that $A_i^k + e \in \mathcal{F}_i$. Let $B_j = \{e : e \in A_j^k, A_i^k + e \in \mathcal{F}_i\}$ and $B_j^* \in \arg\max\{|S| : S \subseteq B_j, A_i^k \cup S \in \mathcal{F}_i\}$. We will show $|B_j| \leq |A_i^k| + 1$. Assume that $|B_j| \geq |A_i^k| + 2$, j has the priority higher than i at least $|B_j| \geq |A_i^k| + 2$ times, which contradicts the order in the algorithm. Combining the property (3) of the matroid, we have $v_i(A_j^k) = |A_i^k| + |B_j^*| \leq |A_i^k| + |B_j| \leq 2|A_i^k| + 1$. Thus, $\mathbf{A}^k$ is $\frac{1}{2}$-EF1.

Next, we show that $\mathbf{A}^k$ is $\frac{1}{2}$-MMS. Consider a MMS partition $\mathbf{X} = (X_1, \ldots, X_n)$ for agent i of item set T, we assume that the value of agent i's bundle in $\mathbf{A}^k$ is less than half of the maximin share of i, i.e., $v_i(A_i^k) \leq \frac{1}{2} v_i(X_j)$ for any $j \in N$. By the property (3) of the matroid, there exists an item $e \in X_j$ such that $A_i^k + e \in \mathcal{F}_i$. Let $C_j = \{e : e \in X_j, A_i^k + e \in \mathcal{F}_i\}$ and $C_j^* \in \arg\max\{|S| : S \subseteq C_j, A_i^k \cup S \in \mathcal{F}_i\}$. Then, we have $v_i(X_j) = v_i(A_i^k \cup C_j^*) = v_i(A_i^k) + v_i(C_j^*) = |A_i^k| + |C_j^*|$. Since $v_i(A_i^k) = |A_i^k| \leq \frac{1}{2} v_i(X_j)$ holds, we get $|C_j| \geq |C_j^*| > |A_i^k| + 1$. Then, summing up all agents, we have $|\cup_{j \in N} C_j| > n(|A_i^k| + 1)$. At last, we get $|\cup_{j \in N} C_j| = |\{e : e \in \cup_{j \in N} X_j, A_i^k + e \in \mathcal{F}_i\}| = |\{e : e \in T, A_i^k + e \in \mathcal{F}_i\}| = |\cup_{j \in N} B_j| \leq (n-1)(|A_i^k| + 1)$, which contradicts the inequality $|\cup_{j \in N} C_j| > n(|A_i^k| + 1)$. Thus, $\mathbf{A}^k$ is $\frac{1}{2}$-MMS.

Finally, we show that $\mathbf{A}^k$ is $\frac{1}{2}$-USW. We make a slight adjustment to the Algorithm 1: when a new item arrives, if the marginal utility is 0 for all agents, then assign the item to agent 1 without updating the agents' order. Then, we will show that $\mathbf{A}^{k'} = (A_1^k \cup A_0^k, \ldots, A_n^k)$ is the output of the adjusted algorithm. We only need to prove that in each round $s \in [k]$, if the marginal value of the arrived item e_s to A_1^{s-1} is 1, we have $v_i(A_1^{s-1} \cup \{e_s\}) = v_i(A_1^{s-1}) + 1$. By the monotonicity of v_i, we get $v_i(A_1^{s-1} \cup A_0^{s-1} \cup \{e_s\}) \geq v_i(A_1^{s-1} \cup \{e_s\}) = v_i(A_1^{s-1}) + 1 = v_i(A_1^{s-1} \cup A_0^{s-1}) + 1$, implying that the item e_s's marginal value to $A_1^{s-1} \cup A_0^{s-1}$ is 1. Thus, $\mathbf{A}^{k'} = (A_1^k \cup A_0^t, \ldots, A_n^k)$ is the output of the adjusted algorithm. Since $v_1(A_1^k \cup A_0^t) = v_1(A_1^k)$, we have $USW(\mathbf{A}^k) = USW(\mathbf{A}^{k'})$. In other words, we only need to show that $\mathbf{A}^{k'}$ is $\frac{1}{2}$-max-USW. The remaining proof part is based on the idea from (Lehmann, Lehmann, and Nisan 2001). Consider the first item e_1, we assume that e_1 is allocated to agent i^* during the execution of the algorithm and $v_{i^*}(e_1) = 1$, $A_{i^*}^1 = \{e_1\}$ and $A_i^1 = \emptyset$ for $i \neq i^*$. Let $\mathbf{Y} = (Y_1, \ldots, Y_n)$ be an allocation achieving the optimal USW of instance $I = (T^k, N, (v_i)_{i \in N})$, and we assume that e_1 is allocated to agent i^o in Y, $e_1 \in Y_{i^o}$. Next, we construct a new instance $I^{(1)} = (T', N, (v_i^{(1)})_{i \in N})$, where $v_{i^*}^{(1)} = \Delta_{e_1}^{i^*}$ and $v_i^{(1)} = v_i$ for all $i \neq i^*$, $T' = T^k \setminus \{e_1\}$, that is, the first item arriving is e_2, and then input the instance $I^{(1)}$ into the adjusted algorithm. Let initialized order of N be $\pi = (\pi_1, \ldots, \pi_n, \pi_{i^*})$, the initialized allocation be $\mathbf{D}^0 = (D_1^0, \ldots, D_n^0)$, where $D_{i^*} = \{e_1\}$ and $D_i = \emptyset$ for all $i \neq i^*$. Assume that $\mathbf{D}^k = (D_1^k, \ldots, D_n^k)$ is the allocation at the end of round k. We have $\mathbf{D}^k = \mathbf{A}^{k'}$, then $USW(\mathbf{D}^k) = USW(\mathbf{A}^{k'}) = USW(\mathbf{A}^k)$. Now, we construct an allocation $\mathbf{S} = (S_1, \ldots, S_n)$ for instance I' as follows: $S_{i^o} = Y_{i^o} \setminus \{e_1\}$ and $S_i = Y_i$ for all $i \neq i^o$. It holds that $v_{i^o}^{(1)}(S_{i^o}) = v_{i^o}(S_{i^o}) = v_{i^o}(Y_{i^o} \setminus \{e_1\}) = v_{i^o}(Y_{i^o}) - v_{i^o}(e_1) \geq v_{i^o}(Y_{i^o}) - v_{i^*}(e_1)$. By the monotonicity of v_{i^*}, we have $v_{i^*}^{(1)}(S_{i^*}) = v_{i^*}(Y_{i^*} \cup \{e_1\}) - v_{i^*}(e_1) \geq v_{i^*}(Y_{i^*}) - v_{i^*}(e_1)$, and $v_{i^o}^{(1)}(S_{i^o}) = v_{i^o}(Y_{i^o} \setminus \{e_1\}) \geq v_{i^o}(Y_{i^o}) - v_{i^o}(e_1) \geq v_{i^o}(Y_{i^o}) - v_{i^*}(e_1)$, where the first inequality follows from the submodularity and the second inequality follows from $v_{i^o}(e_1) \leq v_{i^*}(e_1)$. Let $\mathbf{Y}^{(1)} = (Y_1^{(1)}, \ldots, Y_n^{(1)})$ be an allocation achieving the optimal USW of instance $I^{(1)}$. It holds that $USW(\mathbf{Y}^{(1)}) \geq USW(\mathbf{S}) = \cup_{i \in N} v_i^{(1)}(S_i) = \cup_{i \in N, i \neq i^*, i \neq i^o} v_i^{(1)}(S_i) + v_{i^o}^{(1)}(S_{i^o}) + v_{i^*}^{(1)}(S_{i^*}) \geq USW(\mathbf{Y}) - 2v_{i^*}(e_1)$. By lemma 1 in (Lehmann, Lehmann, and Nisan 2001), $v_{i^*}^{(1)}$ is also submodular. So, by constructing the new instances $I^{(2)}, \ldots, I^{(k-1)}$ and inducting each round based on the items that arrive, we have $USW(\mathbf{A}^k) = USW(\mathbf{A}^{k'}) = USW(\mathbf{D}^k) = USW(\mathbf{Y}^k) \geq \frac{1}{2} USW(\mathbf{Y})$. □

Next, we show that the above approximation results are the best possible by providing the following upper bounds.

Theorem 3. *For any $\epsilon > 0$, with submodular binary valuation functions, no deterministic online algorithm can achieve*

- $(\frac{1}{2} + \epsilon)$*-EF1 and non-wastefulness,*
- *or* $(\frac{1}{2} + \epsilon)$*-MMS and non-wastefulness,*
- *or* $(\frac{1}{2} + \epsilon)$*-max-USW and non-wastefulness.*

Finally, we give the following corollary, which demonstrates the robustness of our algorithm.

Corollary 1. *For the deterministic allocation of indivisible goods with additive binary valuation functions, the Marginal-Greedy Algorithm (Algorithm 1) computes an NW allocation that satisfies EF1, MMS, and max-USW.*

Additive Personalized Bi-valued Valuations

In this part, we consider the additive personalized bi-valued valuation function, which is frequently discussed in fair division, such as (Ebadian, Peters, and Shah 2022; Aziz et al. 2023). For any agent $i \in N$, her valuation function v_i is additive personalized bi-valued if v_i is additive and $v_i(e) \in \{a_i, b_i\}$ for any $e \in T$, where $0 < a_i \leq b_i$.

It is worth investigating whether an NW allocation satisfying EF1 or MMS is achievable with additive personalized bi-valued valuation functions, as the binary case is a special type of bi-valued function. However, we find that even for two agents with additive personalized bi-valued valuations, an NW allocation satisfying EF1 or MMS cannot be guaranteed. Before moving to the next part, we introduce the following technique used in our algorithm.

Definition 6 (Envy Graph for Goods)**.** *Given an allocation $\mathbf{A}^k$, the corresponding envy graph is defined as $G = (V, E)$, where the vertex set V corresponds to the set of agents N, and a directed edge $(i, j) \in E$ iff agent i envies agent j, i.e., $v_i(A_i^k) < v_i(A_j^k)$. Additionally, the directed cycle in the envy graph is called an envy cycle.*

Two Agents We investigate the setting of additive personalized bi-valued valuations for two agents and establish the optimal non-trivial result for simultaneously approximating both EF1 and MMS. To achieve this objective, we propose the Adapted Envy-Graph Procedure (Algorithm 2), which requires sophisticated and subtle case analysis due to the uncertainty of future item values.

The Adapted Envy-Graph Procedure (AEGP) consists of three sub-algorithms: the Envy-Graph Procedure (EGP), the Preliminary-Breaking-Cycle (PBC), and the Deep-Breaking-Cycle (DBC), where the details are deferred in the extended version. The core structure of our algorithm employs the EGP as the primary procedure. However, the EGP has a critical requirement: the input allocation must be EF1 and cannot contain any envy cycle.

When an envy cycle is detected, we execute the PBC and DBC algorithms as auxiliary procedures in subsequent rounds to allocate future items more effectively. Once the output allocation from either the PBC or DBC algorithm satisfies the EGP algorithm's precondition, the EGP algorithm resumes execution. The execution flow of this integrated process is illustrated in Figure 1.

The primary challenge lies in effectively addressing envy cycles. This issue is straightforward to resolve in offline settings, but becomes significantly more complex in online environments where reallocation is infeasible. To overcome this obstacle, we first introduce the PBC algorithm, which is developed through careful analysis of item values over the next two rounds. Specifically, the PBC algorithm can be executed consecutively for at most two rounds. After these two rounds, the PBC algorithm successfully addresses most cases, achieving EF1 allocation without an envy cycle. However, if the allocation remains be $\frac{1}{2}$-EF1, a more strategic approach is required in subsequent rounds to ensure fairer distribution. To address this limitation, we introduce the DBC algorithm, establishing allocation rules based on the parity of consecutive execution counts. Unlike the PBC algorithm, the DBC algorithm can be executed consecutively across multiple rounds while maintaining the $\frac{1}{2}$-EF1 approximation guarantee, before its output allocation satisfies the precondition of the EGP. In summary, both the PBC and DBC algorithms establish specific allocation rules for future items when an envy cycle emerges. These rules not only ensure EF1 approximation throughout their execution but also continuously monitor whether the EGP precondition can be satisfied and actively work to achieve it when feasible.

Theorem 4. *For the deterministic allocation of indivisible goods for two agents with additive personalized bi-valued valuations, the AEGP algorithm (Algorithm 2) computes an NW allocation that satisfies $\frac{1}{2}$-EF1 and $\frac{1}{3}$-MMS.*

Algorithm 2: Adapted Envy-Graph Procedure

Input: An instance $(T, N, (v_1, v_2))$ with additive personalized bi-valued valuation functions
Output: An NW approximate EF1 and approximate MMS allocation $\mathbf{A}$

Let $\mathbf{A} = (\emptyset, \dots, \emptyset)$;
Initialize $\alpha = 1$; `// Algorithm Indicator.`
Initialize (λ_1, μ_1) and (λ_2, μ_2), where $\lambda_1 = \lambda_2 = 1$ and $\mu_1 = \mu_2 = 0$; `//` λ`: the times of consecutive execution for the PBC or DBC algorithm, and` μ`: the case label in the PBC or DBC algorithm.`
Build an envy graph G for two agents;
when *item $e \in T$ arrives***do**
  if $\alpha = 1$ **then**
    $\mathbf{A}, \alpha \leftarrow \mathsf{EGP}(N, \mathbf{A}, \alpha, e)$;
    if $\alpha = 2$ **then**
      $\mathbf{A}, \alpha \leftarrow \mathsf{PBC}(N, \mathbf{A}, \alpha, e, (\lambda_1, \mu_1))$; `// Envy cycle will appear.`
    end
  else if $\alpha = 2$ **then**
    $\mathbf{A}, \alpha, (\lambda_1, \mu_1) \leftarrow \mathsf{PBC}(N, \mathbf{A}, \alpha, e, (\lambda_1, \mu_1))$;
  else if $\alpha = 3$ **then**
    $\mathbf{A}, \alpha, (\lambda_2, \mu_2) \leftarrow \mathsf{DBC}(N, \mathbf{A}, \alpha, e, (\lambda_2, \mu_2))$;
  end
  Update the envy graph G;
return $\mathbf{A}$;

Next, we list the following two lemmas that are crucial in the proof of Theorem 4.

Lemma 3. *The EGP algorithm computes an NW allocation that is EF1 without an envy cycle.*

Lemma 4. *The PBC or DBC algorithm computes an NW allocation satisfying*

- *$\frac{1}{2}$-EF1,*
- *or EF1 with an envy cycle,*
- *or EF1 without an envy cycle.*

Proof of Theorem 4. In the AEGP, it can be seen that for different values of the algorithm indicator α, we need to choose the corresponding algorithm, i.e., $\alpha = 1$: EGP, $\alpha = 2$: PBC, or $\alpha = 3$: DBC, to execute. By Lemmas 3 and 4, it can be seen that the output allocation is NW and $\frac{1}{2}$-EF1. Regarding MMS, by the implication between EF1 and MMS (Amanatidis, Birmpas, and Markakis 2018), i.e., α-EF1 $\Rightarrow$ $\frac{\alpha}{(n-1)\alpha+1}$-MMS, where n is the number of agents, it is clear that the output allocation is $\frac{\frac{1}{2}}{\frac{1}{2}+1} = \frac{1}{3}$-MMS. □

For an independent interest, in the extended version, we show that if each online item has a deadline of one period, i.e., each arriving item can wait for at most one round to be allocated, the EF1 allocation can be found in the above setting. At last, we present the impossible results to show that our approximation guarantees for EF1 and MMS are

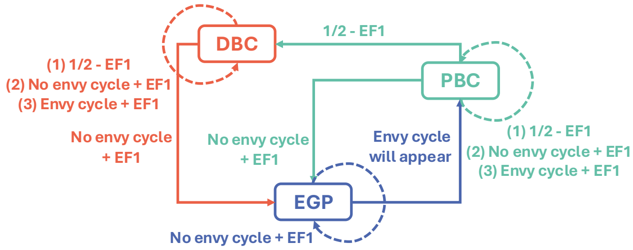

Figure 1: The execution flow of EGP, PBC, and DBC algorithms. The dashed line represents the execution of the corresponding algorithm. The solid line means that when some specific cases occur, we execute another algorithm.

tight, and the approximate EF1 (or MMS) is incompatible with USW.

Theorem 5. *For two agents with additive personalized bi-valued valuation functions, no deterministic online algorithm can compute an NW allocation that is* $(\frac{1}{2}+\epsilon)$*-EF1 or* $(\frac{1}{3}+\epsilon)$*-MMS for any* $\epsilon > 0$*.*

Proposition 1. *Given any* $\alpha, \beta > 0$*, for the deterministic allocation of indivisible goods, no online algorithm can compute an NW allocation satisfying*

- α*-EF1 and* β*-max-USW,*
- *or* α*-MMS and* β*-max-USW,*

even for two agents with additive personalized bi-valued valuations.

The Combination with Additive Binary Valuations In this part, we focus on the setting where only one agent (agent n) has an additive personalized bi-valued valuation function, while all other agents share identical additive binary valuation functions, formed by $(T, N, (v_i)_{i\in N\setminus\{n\}}, v_n)$. We show that an NW allocation that is EF1 and MMS always exists and can be computed by the Adapted-Picking Algorithm (Algorithm 3). The overview of the techniques is to allocate the arriving items that all agents value at non-zero values "evenly", according to the requirement that agent n has the top priority of picking the item with the value of b_n and the lowest priority of picking the item with the value of a_n. In that case, agent n takes advantage of the item options, and then we can guarantee that EF1 and MMS hold simultaneously.

Theorem 6. *For the deterministic allocation of indivisible goods for* $(n-1)$ *agents with identical additive binary valuation functions and one agent with the additive personalized bi-valued valuation function, the Adapted-Picking Algorithm (Algorithm 3) computes an NW, EF1, and MMS allocation.*

Proof Sketch. For EF1, we use induction to show that it holds in every round. The induction step is valid because we allocate each item to a carefully selected agent by the specific rules defined in our algorithm. For MMS, for an arbitrary round, we examine the two types of agents separately. For any agent with identical additive binary valuations, by considering the number of arrived items with value

---

Algorithm 3: Adapted-Picking Algorithm

Input: An instance $(T, N, (v_i)_{i\in N\setminus\{n\}}, v_n)$ with identical $(n-1)$ additive binary valuation functions $(v_1, \ldots, v_n)$ and one additive personalized bi-valued valuation function v_n
Output: An NW, EF1, and MMS allocation $\mathbf{A}$

Let $\mathbf{A} = (\emptyset, \ldots, \emptyset)$;
when *item* $e \in T$ *arrives* **do**
  Let $i_{min} \in \arg\min_{i\in[n-1]} |A_i|$;
  if $v_{i_{min}}(e) = 0$ **then**
    $A_n = A_n \cup \{e\}$;
  else
    if $v_n(e) = b_n$ **then**
      if $v_{i_{min}}(A_n) \le v_{i_{min}}(A_{i_{min}})$ **then**
        $A_n = A_n \cup \{e\}$;
      else
        $A_{i_{min}} = A_{i_{min}} \cup \{e\}$;
      end
    else
      if $v_{i_{min}}(A_n) > v_{i_{min}}(A_{i_{min}})$ **then**
        $A_{i_{min}} = A_{i_{min}} \cup \{e\}$;
      else
        $A_n = A_n \cup \{e\}$;
      end
    end
  end
return $\mathbf{A}$;

---

1 and the EF1 property, we can establish that MMS holds. For agent n with the additive personalized bi-valued valuation, by considering whether she receives an item in the current round and the EF1 property, we can also derive that MMS holds. □

At last, we complement the above result by showing the incompatibility of EF1 (or MMS) and USW.

Proposition 2. *Given any* $\alpha > 0$*, for the deterministic allocation of indivisible goods, no online algorithm can compute an NW allocation satisfying*

- *EF1 and* α*-max-USW,*
- *or MMS and* α*-max-USW,*

even for two agents, where one agent has the additive binary valuation function, and the other agent has the additive personalized bi-valued valuation function.

Conclusion and Future Work

In our paper, we study the fair online allocation of indivisible items and present a series of positive and negative results regarding the existence and approximation results in fairness and efficiency for various valuations and cost functions, which provide a comprehensive view of what can be achieved. For future work, the most interesting direction is to fill the gap in the approximation of EF1 or MMS for additive personalized bi-valued valuation or cost functions for more than two agents, where the current technique cannot be directly generalized due to the complicated envy cycles.

Acknowledgments

Yuanyuan Wang is supported by the Science and Technology Development Fund (FDCT), Macau SAR (file no. 0147/2024/RIA2, 0014/2022/AFJ, 0085/2022/A, 001/2024/SKL, and CG2025-IOTSC).

References

Aleksandrov, M.; Aziz, H.; Gaspers, S.; and Walsh, T. 2015. Online fair division: analysing a food bank problem. In *Proceedings of the 24th International Conference on Artificial Intelligence*, 2540–2546.

Amanatidis, G.; Aziz, H.; Birmpas, G.; Filos-Ratsikas, A.; Li, B.; Moulin, H.; Voudouris, A. A.; and Wu, X. 2023. Fair division of indivisible goods: Recent progress and open questions. *Artificial Intelligence*, 322: 103965.

Amanatidis, G.; Birmpas, G.; and Markakis, E. 2018. Comparing approximate relaxations of envy-freeness. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, 42–48.

Amanatidis, G.; Lolos, A.; Markakis, E.; and Turmel, V. 2025. Online fair division for personalized 2-value instances. In *International Symposium on Algorithmic Game Theory*, 209–227.

Aziz, H.; Lindsay, J.; Ritossa, A.; and Suzuki, M. 2023. Fair Allocation of Two Types of Chores. In *Proceedings of the 22rd International Conference on Autonomous Agents and Multiagent Systems*, 143–151.

Barman, S.; Narayan, V.; and Verma, P. 2023. Fair Chore Division under Binary Supermodular Costs. In *Proceedings of the 22nd International Conference on Autonomous Agents and Multiagent Systems*, 2863–2865.

Barman, S.; and Verma, P. 2021. Existence and Computation of Maximin Fair Allocations Under Matroid-Rank Valuations. In *Proceedings of the 20th International Conference on Autonomous Agents and MultiAgent Systems*, 169–177.

Benabbou, N.; Chakraborty, M.; Igarashi, A.; and Zick, Y. 2021. Finding fair and efficient allocations for matroid rank valuations. *ACM Transactions on Economics and Computation*, 9(4): 1–41.

Benadè, G.; Halpern, D.; and Psomas, A. 2022. Dynamic fair division with partial information. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, 3703–3715.

Benadè, G.; Kazachkov, A. M.; Procaccia, A. D.; Psomas, A.; and Zeng, D. 2024. Fair and efficient online allocations. *Operations Research*, 72(4): 1438–1452.

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

Choo, D.; Fu, W.; Khu, D.; Neoh, T. Y.; Poon, T.-Y.; and Teh, N. 2025. Approximate proportionality in online fair division. *arXiv preprint arXiv:2508.03253*.

Cookson, B.; Ebadian, S.; and Shah, N. 2025. Temporal fair division. In *Proceedings of the 39th AAAI Conference on Artificial Intelligence*, 13727–13734.

Ebadian, S.; Peters, D.; and Shah, N. 2022. How to Fairly Allocate Easy and Difficult Chores. In *Proceedings of the 21st International Conference on Autonomous Agents and Multiagent Systems*, 372–380.

Elkind, E.; Lam, A.; Latifian, M.; Neoh, T. Y.; and Teh, N. 2025. Temporal Fair Division of Indivisible Items. In *Proceedings of the 24th International Conference on Autonomous Agents and Multiagent Systems*, 676–685.

Friedman, E.; Psomas, C.-A.; and Vardi, S. 2015. Dynamic fair division with minimal disruptions. In *Proceedings of the 16th ACM conference on Economics and Computation*, 697–713.

Friedman, E.; Psomas, C.-A.; and Vardi, S. 2017. Controlled dynamic fair division. In *Proceedings of the 18th ACM Conference on Economics and Computation*, 461–478.

Goldman, J.; and Procaccia, A. D. 2015. Spliddit: Unleashing fair division algorithms. *ACM SIGecom Exchanges*, 13(2): 41–46.

He, J.; Procaccia, A.; Psomas, A.; and Zeng, D. 2019. Achieving a fairer future by changing the past. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence*, 343–349.

Hosseini, H.; Huang, Z.; Igarashi, A.; and Shah, N. 2024. Class fairness in online matching. *Artificial Intelligence*, 335: 104177.

Kash, I.; Procaccia, A.; and Shah, N. 2014. No agent left behind: Dynamic fair division of multiple resources. *Journal of Artificial Intelligence Research*, 51: 579–603.

Lee, M. K.; Kusbit, D.; Kahng, A.; Kim, J. T.; Yuan, X.; Chan, A.; See, D.; Noothigattu, R.; Lee, S.; Psomas, A.; et al. 2019. WeBuildAI: Participatory framework for algorithmic governance. *Proceedings of the ACM on human-computer interaction*, 1–35.

Lehmann, B.; Lehmann, D.; and Nisan, N. 2001. Combinatorial auctions with decreasing marginal utilities. In *Proceedings of the 3rd ACM conference on Electronic Commerce*, 18–28.

Li, B.; Li, W.; and Li, Y. 2018. Dynamic fair division problem with general valuations. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, 375–381.

Melissourgos, T.; and Protopapas, N. 2025. Online EFX Allocations with Predictions. *arXiv preprint arXiv:2508.04779*.

Munkres, J. 1957. Algorithms for the assignment and transportation problems. *Journal of the society for industrial and applied mathematics*, 5(1): 32–38.

Neoh, T. Y.; Peters, J.; and Teh, N. 2025. Online Fair Division with Additional Information. *arXiv preprint arXiv:2505.24503*.

Rey, D.; Hammad, A. W.; and Saberi, M. 2023. Vaccine allocation policy optimization and budget sharing mechanism using reinforcement learning. *Omega*, 115: 102783.

Schrijver, A.; et al. 2003. *Combinatorial optimization: polyhedra and efficiency*, volume 24.

Song, J.; Tao, B.; Wang, W.; and Zhang, Y. 2025. Online MMS allocation for chores. *arXiv preprint arXiv:2507.14039*.

Zhou, S.; Bai, R.; and Wu, X. 2023. Multi-agent online scheduling: MMS allocations for indivisible items. In *Proceedings of the 40th International Conference on Machine Learning*, 42506–42516.