Fairness Towards Groups of Agents in the Allocation of Indivisible Items

Nawal Benabbou, Mithun Chakraborty, Edith Elkind, Yair Zick

Fairness Towards Groups of Agents in the Allocation of Indivisible Items

Nawal Benabbou[1], Mithun Chakraborty[2], Edith Elkind[3], and Yair Zick[2]

[1] Sorbonne Université, CNRS, Laboratoire d'Informatique de Paris 6, LIP6 F-75005 Paris, France
nawal.benabbou@lip6.fr
[2] Department of Computer Science, National University of Singapore, Singapore
{mithun,zick}@comp.nus.edu.sg
[3] Department of Computer Science, University of Oxford, United Kingdom
elkind@cs.ox.ac.uk

Abstract. In this paper, we study the problem of matching a set of items to a set of agents partitioned into *types* so as to balance fairness towards the types against overall utility. We extend multiple desirable properties of an allocation of indivisible goods to our model and investigate the possibility and hardness of achieving combinations of these properties, e.g. we prove that maximizing utilitarian social welfare under constraints of typewise envy-freeness up to one item (TEF1) is computationally intractable. In particular, we define a new concept of *waste* for this setting; we show experimentally that augmenting an existing algorithm with a marginal utility maximization heuristic can produce a TEF1 solution with reduced waste, and also provide a polynomial-time algorithm for computing a non-wasteful TEF1 allocation for binary agent-item utilities.

Keywords: Fair Allocation of Indivisible Items · Typewise Envy-freeness up to one item · Non-wasteful Allocation.

1 Introduction

Consider an academic department faced with the task of assigning incoming graduate students to advisers. It has computed a score for each potential advisor-advisee pair and cares about the overall score of the matching; but the faculty is divided into research groups and the department might also wish to achieve a fair distribution of students (or, more correctly, student scores) across these groups. A similar trade-off between efficiency/welfare and fairness might be desirable in other planning/allocation scenarios such as public housing allocation, e.g. agents are potential tenant households and items are flats in public housing blocks, individual agents having utilities over flats and a central authority desiring a fair allocation across groups of agents (in the same vein as the ethnicity quotas of the Singapore Housing and Development Board [3]).

We can model these problems as a variant of the problem of fair allocation of indivisible goods (see e.g. [4]): here, we have an underlying weighted bipartite matching problem, with the nodes on the two sides being called *items* and *agents* for convenience, but the parties we are trying to be fair towards are not individual agents but subsets forming a partition over agents – we call these subsets *types* of agents. An important aspect of this problem is that all agents in a type do not derive utility from all items in the *bundle* allocated to that type (unlike the public goods scenario [12]): within a type, we assign items to agents under matching constraints with no regard to other types. Hence, the parties under consideration end up violating the additive bundle-valuation assumption present in much of recent work. There are some approaches (e.g. [20]) that achieve good fairness guarantees but are agnostic to *how* a party computes its bundle valuation, hence if we use them naïvely, they can result in allocations that are wasteful/inefficient in some way given the structure of our problem. These considerations necessitate novel solution concepts and techniques for our setting.

1.1 Our contributions

We describe a new model of typewise fair allocation and define our desirable properties in Section 2 – in particular, *non-wastefulness* and *typewise envy-freeness up to one item* (TEF1) as well as a marginal envy-based variant of the latter (TMEF1). In Sections 3 and 4, we explore the problems of determining Pareto optimal, TMEF1 allocations and TEF1 allocations that maximize overall sum of weights respectively. In Section 5, we show experimentally that Lipton et al. [20]'s classic algorithm equipped with a simple heuristic can produce TEF1 allocations with significantly reduced waste. Section 6 details a polynomial-time algorithm for computing a non-wasteful TEF1 allocation for binary agent-item utilities. We conclude with future research directions in Section 7.

1.2 Related work

There is a rich body of work on approaches towards the fair allocation of indivisible goods [4,23]. A popular fairness concept is *envy-freeness* [14]. A complete envy-free (EF) allocation may not exist but a relaxation that always does is one that is *envy-free up to one item* (EF1) [6] where any envy towards an agent can be eliminated by removing an item from its bundle. The bounded-envy, polynomial-time algorithm due to Lipton et al. [20] also produces an EF1 allocation for general valuation functions [25].

Extensions of envy-freeness to groups include *strict envy-freeness* [30]; *coalition fairness* or *group envy-freeness* [19] as well as *envy-freeness of an individual/a group towards a group* [28], both under monetary transfers; *group fairness* as defined by Conitzer et al. [8]. Barman et al. [1] recently defined a *groupwise* extension to the *maximin share*-based fairness concept. A major difference of these contributions with our model is that they deal with a σ-algebra of subsets of agents rather than an exogenously defined partition over agents. Notable papers that define fairness with respect to pre-defined groups of multiple players include those by Suksompong along with Manurangsi and Segal-Halevi [22,27,26] where the utility structure is significantly different from ours, and Elzayn et al. [11] whose concepts of utility and fairness are significantly different from ours. Recent work [12,13] has also explored non-envy-based fairness criteria in the allocation of *public goods* under additive valuations.

We must also mention the literature on *statistical fairness* (also called *group fairness*) in the fundamentally different problem domain of classification in machine learning: the equalization of some statistical property of the classifier across groups of data instances based on sensitive/protected attributes ([10,16,17] and references therein); we are interested in fairness notions in terms of subjective valuations of items from the economics/social choice literature.

2 Model and definitions

Throughout the paper, $[r]$ will denote the set $\{1, 2, \cdots, r\}$ for any positive integer r. Our model, an extension of the classic framework of matching on a weighted bipartite graph [21], has the following ingredients:

(i) a set N of n vertices called *agents* partitioned into k types $N_1, \ldots, N_k$,
(ii) a set M of m vertices called *items*,
(iii) a weight/utility $u(i, j) \in \mathbb{R}_+$ for each agent-item edge $(i, j) \in N \times M$, such that for each $i \in N$ (resp. at least one $j \in M$), there is at least one $j \in M$ (resp. each $i \in N$) with $u(i, j) > 0$.

For any $T \subseteq N$ and any $S \subseteq M$, a (T, S)-*matching* is defined as a subset of the edges $T \times S$ such that every vertex in $T \cup S$ is incident on at most one of the edges or, equivalently, as a binary matrix $X = (x_{ij})_{i \in T, j \in S}$ such that for each agent i (resp. item j), there is at most one item j (resp. agent i) with $x_{ij} = 1$, i.e. each item is assigned to at most one agent and each agent is assigned at most one item. The *utilitarian social welfare* $\mathtt{USW}(X)$ (or *total weight*) of a matching X is defined as the sum of the realized utilities of all agents under that matching:

$$\mathtt{USW}(X) \triangleq \sum_{i \in T} \sum_{j \in S} x_{ij} u(i, j).$$

An *optimal matching* is one that maximizes the corresponding utilitarian social welfare. We are interested in a (N, M)-matching that trades off some welfare concept characterizing the entire agent population against some fairness criterion defined with respect to the agent types. More precisely, for every type $p \in [k]$, we are given a *type-value function* $v_p : 2^M \to \mathbb{R}_+$ which quantifies some concept of overall welfare derived by N_p from some *bundle* or subset of items $S \subseteq M$ in terms of the weights $u(i, j)$, $(i, j) \in N_p \times S$. In this paper, we will use the following specific type-value function for every type:

Definition 1 (Utilitarian type-value function) *The* utilitarian type-value *of any type $p \in [k]$ for any bundle $S \subseteq M$ is defined as the total weight of an optimal (N_p, S)-matching:*

$$v_p(S) \triangleq \begin{cases} \max_{X \in \mathcal{X}(N_p, S)} \mathtt{USW}(X), & \text{if } S \neq \emptyset; \\ 0, & \text{otherwise.} \end{cases}$$

where $\mathcal{X}(N_p, S)$ *is the collection of all* (N_p, S)*-matchings.*

We will define the *marginal utility* $\Delta_p(S; j)$ of an item $j \in M$ for a type $p \in [k]$ and a bundle $S \subseteq M$ as:

$$\Delta_p(S; j) \triangleq \begin{cases} v_p(S \cup \{j\}) - v_p(S), & \text{if } j \notin S; \\ v_p(S) - v_p(S \backslash \{j\}), & \text{otherwise.} \end{cases}$$

Given a type p and a bundle S, there can be multiple optimal (N_p, S)-matchings with the same type-value but possibly differing in other efficiency and/or fairness properties (see discussion on Figure 1 at the end of this section) – with this in mind, we define an *allocation* in our setting as follows:

Definition 2 (Allocation) *An* allocation $\mathcal{A}$ *is a collection of bundles* $M_1^{\mathcal{A}}, \cdots, M_k^{\mathcal{A}}$*, such that* $M_1^{\mathcal{A}} \cup \ldots \cup M_k^{\mathcal{A}} \subseteq M$ *and* $M_p^{\mathcal{A}} \cap M_q^{\mathcal{A}} = \emptyset$ *for all* $p, q \in [k]$ *with* $p \neq q$*, along with* an *optimal matching between each type* N_p *and the corresponding bundle* $M_p^{\mathcal{A}}$ *for all* $p \in [k]$*, thereby inducing a unique* (N, M)*-matching* $X^{\mathcal{A}} = (x_{ij}^{\mathcal{A}})_{i \in N, j \in M}$.

We call $M_p^{\mathcal{A}}$ the *allocated bundle* of type p under $\mathcal{A}$ and $M_0^{\mathcal{A}} = M \backslash \cup_{p \in [k]} M_p^{\mathcal{A}}$ the set of *withheld items*. We will sometimes drop the superscript $\mathcal{A}$ when there is no ambiguity. A type p envies a type q if $v_p(M_p^{\mathcal{A}}) < v_p(M_q^{\mathcal{A}})$; p envies q up to ν items, $\nu \in [|M_q^{\mathcal{A}}|]$, if there is a subset $C \subseteq M_q^{\mathcal{A}}$ of size $|C| = \nu$ such that $v_p(M_p^{\mathcal{A}}) \geq v_p(M_q^{\mathcal{A}} \backslash C)$ and, for every subset $C' \subseteq M_q^{\mathcal{A}}$ with $|C'| < \nu$, $v_p(M_p^{\mathcal{A}}) < v_p(M_q^{\mathcal{A}} \backslash C')$. We can analogously define the envy of a type for a bundle (up to any number of items).

With these fundamentals in place, we now define the desiderata of an allocation $\mathcal{A}$ that we investigate in this paper. The first three are concerned with efficiency; the rest are extensions of efficiency-agnostic fairness concepts introduced by Budish [6] and Caragiannis et al. [7] respectively.

Definition 3 (Type-completeness) $\mathcal{A}$ *is* type-complete *if* $\cup_{p \in [k]} M_p^{\mathcal{A}} = M$*; otherwise it is* type-incomplete.[4]

Definition 4 (Waste and non-wastefulness) *An item* $j \in M$ *is said to be* wasted *by an allocation* $\mathcal{A}$ *if it has a positive marginal utility for some type* $p \in [k]$ *(i.e.* $\Delta_p(M_p^{\mathcal{A}}; j) > 0$*) but is either withheld (i.e.* $j \in M_0^{\mathcal{A}}$*) or belongs to the allocated bundle of some type* $q \neq p$ *for which it has zero marginal utility (i.e.* $j \in M_q^{\mathcal{A}}$ *and* $\Delta_q(M_q^{\mathcal{A}}; j) = 0$*).* $\mathcal{A}$ *is called* non-wasteful *if it has no wasted item, and wasteful otherwise.*

Definition 5 (Typewise Pareto optimality) *Allocation* $\mathcal{A}_1$ *is said to* typewise Pareto dominate *another allocation* $\mathcal{A}_2$ *if* $v_p(M_p^{\mathcal{A}_1}) \geq v_p(M_p^{\mathcal{A}_2})$ *for all types* $p \in [k]$ *and* $v_p(M_p^{\mathcal{A}_1}) > v_p(M_p^{\mathcal{A}_2})$ *for some type* $p \in [k]$*. An allocation that is not typewise Pareto dominated by any other allocation is* typewise Pareto optimal.

[4] Type-completness does not preclude an item $j \in M_p^{\mathcal{A}}$ remaining unassigned in the $(N_p, M_p^{\mathcal{A}})$-matching.

Definition 6 (Typewise envy-freeness up to one item) *Allocation $\mathcal{A}$ is* typewise envy-free up to one item *(TEF1) if for any two types $p, q \in [k]$, p either does not envy q or envies q up to one item, i.e. there exists an item $j \in M_q^{\mathcal{A}}$ such that $v_p(M_p^{\mathcal{A}}) \geq v_p(M_q^{\mathcal{A}} \backslash \{j\})$.*

Definition 7 (Typewise marginal envy-freeness up to one item) *Allocation $\mathcal{A}$ is* typewise marginally envy-free up to one item *(TMEF1) if for any $p, q \in [k]$, there is an item $j \in M_q^{\mathcal{A}}$ such that $v_p(M_p^{\mathcal{A}}) \geq v_p(M_p^{\mathcal{A}} \cup M_q^{\mathcal{A}} \backslash \{j\}) - v_p(M_p^{\mathcal{A}})$.*

We are now ready to formulate and analyze specific problems that approach 'good' allocations in different ways. But first, we will provide a problem instance that we will use as a running example throughout the rest of the paper.

Example 1 *Consider the problem depicted in Figure 1 and the bundles $M_1 = \{1, 2, 6\}$, $M_2 = M \backslash M_1$: there is a unique optimal (N_2, M_2)-matching with $3, 4, 5$ assigned to b_1, b_2, b_3 respectively; but there are two optimal (N_1, M_1)-matchings in both of which 2 is assigned to a_2: if 1 is assigned to a_1 and 6 remains unassigned, then we have a wasteful allocation since 6 could be assigned to b_4 so that $\Delta_2(M_2; 6) = 1$; but if 6 is assigned to a_1 instead, the allocation is non-wasteful since no agent in N_2 has a positive utility for item 1.*

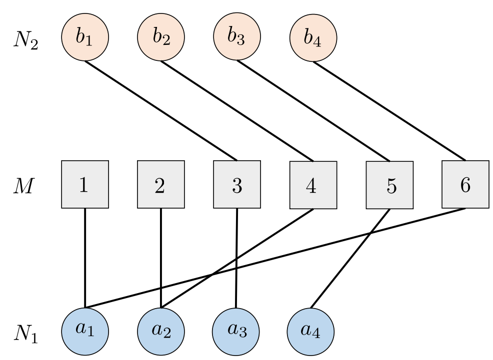

Fig. 1: Our running example: 2 types $N_1 = \{a_1, \ldots, a_4\}$ and $N_2 = \{b_1, \ldots, b_4\}$; items $M = [6]$; $u(i, j)$ is 1 if there is an edge between agent i and item j, and 0 otherwise.

3 Typewise Pareto optimal TMEF1 allocation

In this section, we will adapt a result of Caragiannis et al. [7] to our setting. To that end, we first state without proof a property of our type-value function (Definition 1) that is part of the folklore of weighted bipartite matching.

Theorem 1. *The utilitarian type-value function $v_p(S)$ is a non-additive, monotone submodular function of $S \in 2^M$.*

Moreover, we define the *Nash type-welfare* of an allocation $\mathcal{A}$ as the product of the type-values, i.e. $\prod_{p=1}^{k} v_p(M_p^{\mathcal{A}})$.

Theorem 2. *Every type-complete allocation that maximizes the Nash type-welfare is typewise Pareto optimal and TMEF1.*

Proof. An "allocation", as defined in [7] as a partition of the set of items among all agents, translates to a type-complete allocation in our setting; also, since each $v_p(\cdot)$ is monotone submodular by Theorem 1, we can use the natural extension of Theorem 3.5 of [7] to conclude that *every* maximum Nash type-welfare, type-complete allocation is TMEF1 as well as typewise Pareto optimal. □

The next result establishes that non-wastefulness is a weakening of typewise Pareto optimality in our setting.

Lemma 1. *Any typewise Pareto optimal allocation is non-wasteful but the converse is not true.*

Proof. If an allocation $\mathcal{A}$ had a wasted item j with positive marginal utility for a type q, then, by the definition of waste, we could augment $M_q^{\mathcal{A}}$ with j for an improved type-value without reducing any other type-value, resulting in an allocation that Pareto dominates $\mathcal{A}$. For the converse, Example 1 provides a counterexample: Take again the allocation $\mathcal{A}_1$ with bundles $M_1^{\mathcal{A}_1} = \{1, 2, 6\}$, $M_2^{\mathcal{A}_1} = M \backslash M_1^{\mathcal{A}_1}$, item 1 remaining unassigned and $2, \ldots, 6$ assigned to a_2, b_1, b_2, b_3, a_1 respectively. This is non-wasteful (and incidentally also TEF1) but is Pareto dominated by allocation $\mathcal{A}_2$ with bundles $M_1^{\mathcal{A}_2} = \{1, 2, 3\}$, $M_2^{\mathcal{A}_2} = M \backslash M_1^{\mathcal{A}_2}$ since $v_1(M_1^{\mathcal{A}_2}) = 3 > 2 = v_1(M_1^{\mathcal{A}_1})$ and $v_2(M_2^{\mathcal{A}_2}) = v_2(M_2^{\mathcal{A}_1}) = 3$. □

This result implies, in conjunction with Theorem 2, that a maximum Nash type-welfare, type-complete allocation is also non-wasteful. In spite of the above existence results, maximizing Nash welfare with indivisible items is known to be hard in general, and the above guarantees may break down for constant-factor approximations [7]. Moreover, marginal envy-freeness up to one item is a relatively new fairness concept that is hard to explain and not extensively used as yet. Hence, in subsequent chapters, we will focus on efficient TEF1 allocations.

4 Assignment under TEF1 constraints

We first study the problem of finding a TEF1 allocation $\mathcal{A}$ that maximizes the sum of weights of the induced matching:

$$\mathtt{USW}(X^{\mathcal{A}}) \triangleq \sum_{i \in N} \sum_{j \in M} x_{ij}^{\mathcal{A}} u(i,j) = \sum_{p \in [k]} v_p(M_p^{\mathcal{A}}).$$

This is equivalent to the *assignment* problem [24] under TEF1 constraints specified in Definition 6. We define the decision version of the problem as follows:

Definition 8 (AssignTEF1) *An instance of the Assignment under TEF1 constraints (*AssignTEF1*) problem is given by parameters* (i) *to* (iii) *of Section 2 as well as a value* $U \in \mathbb{R}_+$*; it is a 'yes'-instance iff it admits a TEF1 allocation* $\mathcal{A}$ *with a utilitarian social welfare at least* U*.*

A TEF1 allocation always exists under our Definition 2 (see Section 5 for further details); we prove next that it is hard to compute one with the maximum `USW`.

Theorem 3. *The AssignTEF1 problem is NP-complete, even with only* 3 *types.*

Proof. The problem is in NP: given an allocation $\mathcal{A}$, we need to evaluate k type-value function, solving the polynomial-time unconstrained assignment problem (see e.g. [18]) each time, and can hence verify that $\mathcal{A}$ satisfies all requirements in polynomial time.

We will now describe a polynomial-time reduction to AssignTEF1 from the NP-complete *partition* problem [15]. An instance of the latter is given by a set $S = \{s_j\}_{j \in [l]}$ of l positive integers that sum to σ; it is a 'yes'-instance iff S can be partitioned into two subsets S_1 and S_2 such that both sum to $\sigma/2$. Given an instance of the partition problem, we construct an AssignTEF1 instance as follows. We have a set of $l+2$ items $M = [l+2]$ and a set of $2l+4$ agents N partitioned into $k = 3$ types $N_1 = \{a_i\}_{i \in [l+1]}$, $N_2 = \{b_i\}_{i \in [l+1]}$, and $N_3 = \{c_1, c_2\}$. The utilities are given by $u(a_j, j) = u(b_j, j) = s_j$, $\forall j \in [l]$; $u(a_{l+1}, j) = u(b_{l+1}, j) = \sigma/2$, $\forall j \in \{l+1, l+2\}$; $u(c_1, l+1) = u(c_2, l+2) = \kappa$ for an arbitrarily large constant $\kappa > \sigma$; $u(i, j) = 0$ for every other (i, j)-pair. Finally, let $U = 2\kappa + \sigma$.

First, we prove that, for any 'yes'-instance of the partition problem, so is the corresponding AssignTEF1 instance we constructed. Given the two parts S_1 and S_2 of S as above, consider the bundles $M_1 = \{j \in M : s_j \in S_1\}$, $M_2 = \{j \in M : s_j \in S_2\}$, and $M_3 = \{l+1, l+2\}$ allocated to N_1, N_2, and N_3 respectively, with no withheld items; evidently, assigning item j to agent a_j (resp. b_j) for every s_j in S_1 (resp. S_2), $l+1$ to c_1, and $l+2$ to c_2 constitutes the unique optimal matching between each type and its allocated bundle, inducing an allocation $\mathcal{A}$ (Definition 2). We want to prove that $\mathcal{A}$ is TEF1. Note that for the above utilities, $v_3(M_3) = 2\kappa > 0 = v_3(M_q)$ for all $q \in \{1, 2\}$; $v_1(M_1) = v_2(M_1) = \sum_{s_j \in S_1} s_j = \sigma/2$ and also $v_2(M_2) = v_1(M_2) = \sum_{s_j \in S_2} s_j = \sigma/2$ from the definition of a 'yes'-instance of the partition problem; $v_1(M_3 \setminus \{j\}) = v_2(M_3 \setminus \{j\}) = \sigma/2$ for all $j \in M_3$. Hence, each type envies any other type up to at most one item under $\mathcal{A}$. Finally, $\mathtt{USW}(X^{\mathcal{A}}) = \sigma/2 + \sigma/2 + 2\kappa = 2\kappa + \sigma = U$.

Next, we prove that, assuming our constructed AssignTEF1 instance to be a 'yes'-instance, so is the partition instance. Let $\mathcal{A}$ be an allocation verifying all desiderata. Since $\kappa > \sigma$ and the maximum sum of realized utilities that can be achieved from items $j \in [l]$ is σ, the only way for $\mathtt{USW}(X^{\mathcal{A}})$ to be at least $U = 2\kappa + \sigma$ is to have $\{l+1, l+2\} \subseteq M_3^{\mathcal{A}}$ with c_1 (resp. c_2) assigned to $l+1$ (resp. $l+2$). Moreover, since the other items must contribute a sum of realized utilities at least σ, the utility structure implies that each $j \in [l]$ must be assigned to either a_j or b_j; hence, $M_3 = \{l+1, l+2\}$, there are no withheld items, and $v_1(M_1^{\mathcal{A}}) + v_2(M_2^{\mathcal{A}}) = \sigma$. Now consider the sets $S_1 = \{s_j \in S : j \in M_1^{\mathcal{A}}\}$ and $S_2 =$

$\{s_j \in S : j \in M_2^{\mathcal{A}}\}$: It is evident that they form a partition of S such that the sum of the values in S_1 (resp. S_2) equals $v_1(M_1^{\mathcal{A}})$ (resp. $v_2(M_2^{\mathcal{A}})$). Since $\mathcal{A}$ is TEF1, we must have $v_1(M_1^{\mathcal{A}}) \geq v_1(M_3^{\mathcal{A}} \setminus \{j\})$ for some $j \in M_3^{\mathcal{A}}$; but, from our utility structure, $v_1(M_3^{\mathcal{A}} \setminus \{j\}) = \sigma/2$ for all $j \in M_3^{\mathcal{A}}$ and so the inequality $v_1(M_1^{\mathcal{A}}) \geq \sigma/2$ holds. Arguing similarly, $v_2(M_2^{\mathcal{A}}) \geq \sigma/2$. But since $v_1(M_1^{\mathcal{A}}) + v_2(M_1^{\mathcal{A}}) = \sigma$, then $\sum_{s_j \in S_1} s_j = v_1(M_1^{\mathcal{A}}) = \sum_{s_j \in S_2} s_j = v_2(M_2^{\mathcal{A}}) = \sigma/2$. □

Since $v_p(\cdot)$ is a particular submodular function, Theorem 3 implies the following result that applies to the traditional indivisible item allocation setting where each agent receives a bundle.

Corollary 1 *For submodular agent valuation functions over bundles, it is NP-hard to compute the EF1 allocation that maximizes the sum of valuations.*

One might conjecture that the maximum-`USW` TEF1 allocation is non-wasteful. But the following surprising result belies this intuition, and raises the question: Does a non-wasteful, TEF1 allocation always exist?

Proposition 1 *The TEF1 allocation that maximizes the utilitarian social welfare may waste items, even in a problem instance that admits a non-wasteful TEF1 allocation.*

Proof. Consider a problem with items $M = \{1, 2, 3, 4, 5\}$, agents $N = \{1, 2, 3, 4, 5\}$ divided into 2 types $N_1 = \{1, 2\}$ and $N_2 = \{3, 4, 5\}$, and utilities $u(i, 1) = u(i, 2) = 2$, $u(i, 3) = u(i, 4) = 4$, $u(i, 5) = 1$ for every $i \in N_1$; $u(i, 1) = u(i, 2) = 0$, $u(i, 3) = u(i, 4) = 8$, $u(i, 5) = 1$ for every $i \in N_2$. Any allocation with bundles $M_1 = \{1, 2\}$ and $M_2 = \{3, 4\}$, and no other allocation, maximizes `USW` under TEF1 constraints with `USW` = 20. But, such an allocation is wasteful since item 5 is withheld although $\Delta_2(M_2; 5) = 1$. However, any allocation with bundles $M'_1 = \{1, 3\}$ and $M'_2 = \{2, 4, 5\}$ is non-wasteful and TEF1 but has `USW` = 17. □

5 TEF1 allocation with waste reduction

In our quest for a TEF1 allocation with no (or, at least, low) waste, we note that it is possible to obtain a type-complete TEF1 allocation in polynomial time by a natural extension (called **L** hereafter) of the algorithm due to Lipton et al. [20]: Iterate over the items $j \in M$, giving item j to a type, say p, that is currently not envied by any other type for its current bundle M_p; compute an optimal matching with the augmented bundle $M_p \cup \{j\}$; construct the *envy graph* where there is a directed edge from a type q to a type r whenever q envies r and eliminate any cycle in this graph by transferring the bundle of every type on this cycle to its predecessor on this cycle (to ensure that there is an unenvied type in each iteration), followed by re-matching within each such type. Although no item is withheld, it is possible for the final allocation to be wasteful: an item may be allocated to a type which has zero marginal utility for it or may *become* unassigned after a bundle is transferred between types.

One heuristic that could reduce waste is to allocate the item to the unenvied type that has the maximum marginal utility for it, breaking further ties uniformly at random, rather than to an arbitrary unenvied type – we call **L**, augmented with this heuristic, **H**. Unfortunately, Example 1 shows that, **H** can be wasteful in general. Consider the order $1, 2, \cdots, 6$ over items: 1 and 2 are obviously allocated to N_1 while, depending on how ties are broken, $3, 4, 5$ can all be given to N_2. With this allocation of item 5, envy appears for the first time and N_1 is the only unenvied type. Hence, 6 must go to N_1 although $\Delta_1(\{1, 2\}; 6) = 0$ and is wasted. Notice further that if 6 were allocated to N_2, it would increase N_2's own type-value but make N_1 envy N_2 up to 2 items although N_1 does not want 6 in conjunction with its current bundle! This is especially disappointing since Example 1 admits three non-wasteful TEF1 allocations, which are also typewise Pareto optimal and maximize `USW`, with bundles $M_1^* = \{1, 2, 3\}$, $\{1, 2, 5\}$, or $\{1, 2, 3, 5\}$, and $M_2^* = M \backslash M_1^*$ (each resulting in a unique optimal matching for each type).

Nevertheless, To see how the marginal utility maximization heuristic performs in practice, we experimentally compared procedures **L** and **H** using the percentage of items wasted as our performance metric. We simulated two sets of problem instances with $n = 100$ agents partitioned into $k = 3$ types: UNEQUAL: $|N_1| = 74$, $|N_2| = 13$, $|N_3| = 13$[5]; EQUAL: $|N_p| \approx n/k$ for all types $p \in [k]$. For each, we used $m \in \{50, 100\}$ items; for each agent, we sampled m numbers uniformly at random from $[0, 1]$ and normalized them to generate utilities for all m items. We report results averaged over 100 runs each.

We find that **L** wastes 39% (resp. 13%) of the items on average with $m = 100$ (resp. $m = 50$) for UNEQUAL, and, for **Equal**, 0.005% of the items with $m = 100$ and *no* item with $m = 50$; **H** gives non-wasteful solutions for all instances. Thus, we can conclude that the performance of **L** the natural extension of [20] strongly depends on parameters such as type proportions and the number of items, whereas augmenting it with the heuristic under consideration gives surprisingly good results over a variety of input instances.

6 Binary utilities: non-wasteful TEF1 allocation

In this section, we fill focus on the *binary utility model*: $u(i, j) \in \{0, 1\}$, $\forall i \in N$, $\forall j \in M$. This captures scenarios where each agent either approves or disapproves of an item but does not distinguish among its approved items. There exists prior work on fair allocation algorithms producing allocations with binary item utilities [2] but most assume additive bundle valuations.

Theorem 4. *For any problem instance with a binary utility model, there exists a non-wasteful TEF1 allocation that can be computed in polynomial time.*

Our proof is constructive: We provide and analyze an allocation algorithm for the problem (Algorithm 1). Like Lipton et al. [20], we iterate over the items

[5] These numbers roughly follow the proportions of Chinese, Malay, and Indian/Other residents of Singapore according to the 2010 census report [9].

Algorithm 1: PMURR($\{N_p\}_{p\in[k]}, M, (u(i,j))_{i\in N, j\in M}$)

```
Withheld set M_0 ← M;
Temporary set M_T ← ∅;
Allocated bundles M_p ← ∅ ∀p ∈ [k].
repeat
    for j ∈ M_0 do
        if Δ_p(M_p; j) = 0 ∀p ∈ [k] then
            M_0 ← M_0\{j}; M_T ← M_T ∪ {j}.
        else
            Find a type p such that Δ_p(M_p; j) > 0.
            M_0 ← M_0\{j}; M_p ← M_p ∪ {j}.
            if some type envies p up to more than 1 item then
                /*Revocation and reallocation*/
                repeat
                    Find types q, r such that r envies q up to more than 1 item.
                    Find item j' ∈ M_q such that Δ_r(M_r; j') = 1.
                    M_q ← M_q\{j'}; M_r ← M_r ∪ {j'}.
                until no type envies another up to more than 1 item.
    end
    M_0 ← M_T; M_T ← ∅.
until M_0 = ∅ or Δ_p(M_p; j) = 0 ∀j ∈ M_0, ∀p ∈ [k].
```

but decide to augment a type's bundle with an item not based on (the absence of) envy towards it but on the marginal utility of the type for the item. Let us call any algorithm that starts with empty bundles and follows the principle of augmenting the current bundle M_p of a type p with an item j *only if* p has **p**ositive **m**arginal **u**tility (i.e. $\Delta_p(M_p; j) > 0$) a PMU algorithm.

Proposition 2 *Under the binary utility model, we have $v_p(S) \leq \min\{|N_p|, |S|\}$, $\forall p \in [k]$, $\forall S \subseteq M$. In particular, any type's value for its allocated bundle at any stage of a PMU algorithm is equal to the cardinality of the bundle, i.e. $v_p(M_p) = |M_p|$, $\forall p \in [k]$.*

Proof. The first part follows directly from the fact, for binary utilities, that any item can contribute either 1 or 0 to the type-value of any bundle, i.e. $\Delta_p(S; j) \in \{0, 1\}$ for any $p \in [k]$, $S \subseteq M$, and $j \notin S$. For the second part, note that each type starts with an empty bundle (hence zero type-value) and increases its type-value by 1 every time it acquires an item under a PMU algorithm; moreover, since all positive utilities are equal, the only way for an item j to improve the type-value of p is to ensure that every item is assigned to an agent in every $(N_p, M_p \cup \{j\})$-matching – hence, if an item is revoked, both the bundle-size and type-value diminish by 1. □

Corollary 2 *A type $p \in [k]$ with $|M_p| = |N_p|$ under a PMU algorithm cannot envy any bundle $S \subseteq M$ since $v_p(S) \leq |N_p|$.*

If after receiving a new item, a type is still envied by all others up to at most 1 item, no further action is necessary. But a PMU approach by itself cannot ensure that no type will start envying the recipient up to more than 1 item if the latter was already envied up to 1 item. If envy does exceed the acceptable limit, we execute a special **r**evocation and **r**eallocation (RR) subroutine repeatedly until we restore the TEF1 property – hence, we call our algorithm PMURR. The functioning of the RR subroutine depends on the following result:

Proposition 3 *At any stage of a PMU algorithm, if type p envies type q up to ν or more items, then $|M_q| \geq |M_p| + \nu$.*

Proof. It follows from the definition of envy up to ν (or more) items: for any subset $C \subseteq M_q$ with $|C| = \nu - 1$, $v_p(M_p) < v_p(M_q \backslash C)$. But, from Proposition 2 (and since $C \subseteq M_q$), we must have $v_p(M_q \backslash C) \leq |M_q \backslash C| = |M_q| - |C| = |M_q| - \nu + 1$. Combining this with the fact that $v_p(M_p) = |M_p|$ (Proposition 2), we get $|M_p| < |M_q| - \nu + 1$, i.e. $|M_p| \leq |M_q| - \nu$. □

Corollary 3 *Type p can envy another type q only if q has an allocated bundle of a larger size than p which, in turn, implies that there can be no cycles in the envy graph among types for binary utilities under a PMU algorithm.*

The following lemma proves that the existence of envy is sufficient for having an item that can be revoked and reallocated.

Lemma 2. *At any stage of a PMU algorithm, if type p envies type q, then there is an item $j \in M_q$ which has a positive marginal utility for p, i.e. $\Delta_p(M_p; j) = 1$.*

Proof. By Proposition 2 and Corollary 2, if N'_p is the subset of agents who are assigned items in the current (N_p, M_p)-matching, then we must have $v_p(M_p) = |M_p| = |N'_p| < |N_p|$ for p to envy q. Now, if N^q_p is the subset of agents who would be assigned items in a (N_p, M_q)-matching, we must have $|N^q_p| = v_p(M_q) > v_p(M_p) = |N'_p|$, so that $|N^q_p \backslash N'_p| \geq |N^q_p| - |N'_p| > 0$. Since $N^q_p \backslash N'_p \subseteq N_p \backslash N'_p$, each agent in $N^q_p \backslash N'_p$ is assigned no item under the current matching but has a positive utility for a distinct item in M_q. □

Evidently, a revocation from q (resp. a reallocation to r) decrements (resp. increments) its type-value. But, it is not immediately obvious how it affects all envy relations (other types that already envied q or r up to 1 or more items, types that q might start envying up to 1 or more items, etc.) and whether we could trigger a never-ending chain reaction. Our final lemma dispels such doubts.

Lemma 3. *The revocation and reallocation subroutine produces a TEF1 allocation in a polynomial number of steps.*

Proof. First note that computing a type-value for any bundle is polynomial-time and so is checking whether r envies q up to more than 1 item (i.e. $v_r(M_r) < v_q(M_q \backslash \{j\})$ for some $j \in M_q$). Now, for an iteration of the RR subroutine to occur, we must have a type r envying a type q up to 2 or more items, so that $|M_q| \geq |M_r| + 2$ by Proposition 3. For any collection of bundles $\overline{M} = \{M_1, \ldots M_2\}$, define the potential function $\Phi(\overline{M}) \triangleq \sum_{p \in [k]} |M_p|^2$, and let $\overline{M}' =$

$\{M'_1, \dots M'_2\}$ be the collection of bundles after revocation from q and reallocation to r. Hence we have $|M'_q| = |M_q| - 1$ and $|M'_r| = |M_r| + 1$ (by Proposition 2) but $|M'_p| = |M_p|$ for every other type p. Thus, upon simplification,

$$\begin{aligned}\Phi(\overline{M}') - \Phi(\overline{M}) &= |M'_q|^2 - |M_q|^2 + |M'_r|^2 - |M_r|^2 = 2(1 + |M_r| - |M_q|) \\ &\leq 2(1-2) = -2,\end{aligned}$$

i.e. Φ strictly decreases with each RR iteration, and obviously lies between 0 and m^2 (since $\sum_{p \in [k]} |M_p| \leq m$). Hence, RR terminates after a polynomial number of iterations; by the stop criterion, the final allocation is TEF1. □

We are now ready to prove the main result of this section.

Proof (Theorem 4).

Non-wastefulness: By construction, no bundle-augmentation and revocation-and-reallocation allow an item in an allocated bundle to remain unassigned; by Corollary 3, no envy cycles are ever formed in the envy graph among types, hence bundles are never passed between types (unlike in the decycling procedure of [20]) and so no item once assigned can become unassigned (or transferred to the temporary set). However, within the **for** loop, the revocation(s) might result in an agent becoming unassigned who has a positive utility for a currently withheld item that was put in the temporary set because it previously had zero marginal utility for each type (hence this item now becomes a wasted item). This necessitates the outer **repeat** loop whose stop criterion ensures that there are no wasted items in the withheld set at the end of the algorithm.

TEF1 property: Lemma 3 ensures that the allocation is TEF1 after every iteration of the **for** loop, hence upon termination of the algorithm.

Polynomial time-complexity: Computing $v_p(\cdot)$ and $\Delta_p(\cdot)$, verifying envy up to more than 1 item, and finding an item to revoke and reallocate are all polynomial-time; Lemma 3 ensures that each iteration of the **for** loop takes polynomial time. With each iteration of the outer **repeat** loop, the size of the withheld set, which starts at m, strictly decreases (if it does not, it must mean that each withheld item has zero marginal utility for every type – a stop criterion), hence we have a linear number of iterations. □

7 Discussion and future work

We have introduced and investigated new fairness and efficiency concepts for the allocation of indivisible goods to agents of pre-defined types. Evidently, the most important open question is whether a non-wasteful TEF1 allocation exists for every problem instance with arbitrary (non-negative) agent-item utilities. Important properties (such as the equality of bundle size and type-value) do not carry over from binary to arbitrary utilities, hence extensions of our PMURR algorithm to general utility models remain elusive.

It will also be interesting to deduce a theoretical upper bound on the number of wasted items for heuristic approaches such as the one in Section 5.

We considered one type-value function (Definition 1) here, which already posed interesting challenges. A natural alternative, the *average utilitarian type-value* $\hat{v}_p(S) \triangleq v_p(S)/|N_p|$ is equivalent to $v_p(S)$ for all intents and purposes in this paper. More complex functions such as OWA operators [29] or those that address fairness within a type (e.g. [5]) merit further analysis.

Other possible directions for future research include non-envy-based fairness concepts (egalitarian type-welfare, proportionality, maximin share etc.) as well as strategic implications of typewise fair allocation algorithms.

References

1. Barman, S., Biswas, A., Krishnamurthy, S.K., Narahari, Y.: Groupwise maximin fair allocation of indivisible goods. In: Proceedings of the 32nd AAAI Conference on Artificial Intelligence (AAAI). pp. 917–924 (2018)
2. Barman, S., Krishnamurthy, S.K., Vaish, R.: Greedy algorithms for maximizing Nash social welfare. In: Proceedings of the 17th International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS). pp. 7–13 (2018)
3. Benabbou, N., Chakraborty, M., Xuan, V.H., Sliwinski, J., Zick, Y.: Diversity constraints in public housing allocation. Proceedings of the 17th International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS) pp. 973–981 (2018)
4. Bouveret, S., Chevaleyre, Y., Maudet, N.: Fair allocation of indivisible goods. In: Brandt, Felix and Conitzer, Vincent and Endriss, Ulle and Lang, Jérôme and Procaccia, Ariel D. (ed.) Handbook of Computational Social Choice, chap. 12, pp. 284–310. Cambridge University Press (2016)
5. Bouveret, S., Lang, J.: A general elicitation-free protocol for allocating indivisible goods. In: Proceedings of the 20th International Joint Conference on Artificial Intelligence (IJCAI). pp. 73–78 (2011)
6. Budish, E.: The combinatorial assignment problem: Approximate competitive equilibrium from equal incomes. Journal of Political Economy **119**(6), 1061–1103 (2011)
7. Caragiannis, I., Kurokawa, D., Moulin, H., Procaccia, A.D., Shah, N., Wang, J.: The unreasonable fairness of maximum Nash welfare. In: Proceedings of the 17th ACM Conference on Economics and Computation (EC). pp. 305–322. ACM (2016)
8. Conitzer, V., Freeman, R., Shah, N., Vaughan, J.W.: Group fairness for indivisible good allocation. In: Proceedings of the 33rd AAAI Conference on Artificial Intelligence (AAAI) (2019), to appear
9. Department of Statistics, Singapore: Singapore 2010 Census: Key Indicators of the Resident Population (2010)
10. Dwork, C., Hardt, M., Pitassi, T., Reingold, O., Zemel, R.: Fairness through awareness. In: Proceedings of the 3rd Innovations in Theoretical Computer Science Conference (ITCS). pp. 214–226. ACM (2012)
11. Elzayn, H., Jabbari, S., Jung, C., Kearns, M., Neel, S., Roth, A., Schutzman, Z.: Fair algorithms for learning in allocation problems. In: Proceedings of the 2nd ACM Conference on Fairness, Accountability, and Transparency (ACM FAT*) (2019), to appear
12. Fain, B., Goel, A., Munagala, K.: The core of the participatory budgeting problem. In: Proceedings of the 12th Conference on Web and Internet Economics (WINE). pp. 384–399 (2016)

13. Fain, B., Munagala, K., Shah, N.: Fair allocation of indivisible public goods. In: Proceedings of the 19th ACM Conference on Economics and Computation (EC). pp. 575–592 (2018)
14. Foley, D.K.: Resource allocation in the public sector. Yale Economic Essays **7**, 45–98 (1967)
15. Garey, M.R., Johnson, D.S.: Computers and Intractability: A Guide to the Theory of NP-Completeness. W. H. Freeman & Co. (1979)
16. Hardt, M., Price, E., Srebro, N., et al.: Equality of opportunity in supervised learning. In: Advances in neural information processing systems. pp. 3315–3323 (2016)
17. Kearns, M., Neel, S., Roth, A., Wu, Z.S.: Preventing fairness gerrymandering: Auditing and learning for subgroup fairness. In: Proceedings of the 35th International Conference on Machine Learning (ICML). pp. 2564–2572 (2018)
18. Kuhn, H.W.: The Hungarian method for the assignment problem. Naval Research Logistics (NRL) **2**(1-2), 83–97 (1955)
19. Lahaie, S., Parkes, D.C.: Fair package assignment. In: Proceedings of the 1st International ICST Conference on Auctions, Market Mechanisms and Their Applications (AMMA). p. 92 (2009)
20. Lipton, R.J., Markakis, E., Mossel, E., Saberi, A.: On approximately fair allocations of indivisible goods. In: Proceedings of the 5th ACM Conference on Electronic Commerce (EC). pp. 125–131. ACM (2004)
21. Lovász, L., Plummer, M.D.: Matching theory, vol. 367. American Mathematical Society (2009)
22. Manurangsi, P., Suksompong, W.: Asymptotic existence of fair divisions for groups. Mathematical Social Sciences **89**, 100–108 (2017)
23. Markakis, E.: Approximation algorithms and hardness results for fair division with indivisible goods. In: Endriss, U. (ed.) Trends in Computational Social Choice, chap. 12, pp. 231–247. AI Access (2017)
24. Munkres, J.: Algorithms for the assignment and transportation problems. Journal of the Society for Industrial and Applied Mathematics **5**(1), 32–38 (1957)
25. Plaut, B., Roughgarden, T.: Almost envy-freeness with general valuations. In: Proceedings of the 29th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA). pp. 2584–2603. Society for Industrial and Applied Mathematics (2018)
26. Segal-Halevi, E., Suksompong, W.: Democratic fair allocation of indivisible goods. In: Proceedings of the 27th International Joint Conference on Artificial Intelligence (IJCAI). pp. 482–488 (2018)
27. Suksompong, W.: Approximate maximin shares for groups of agents. Mathematical Social Sciences **92**, 40–47 (2018)
28. Todo, T., Li, R., Hu, X., Mouri, T., Iwasaki, A., Yokoo, M.: Generalizing envy-freeness toward group of agents. In: Proceedings of the 22nd International Joint Conference on Artificial Intelligence (IJCAI). pp. 386–392 (2011)
29. Yager, R.R.: On Ordered Weighted Averaging aggregation operators in multicriteria decision making. IEEE Transactions on systems, Man, and Cybernetics **18**(1), 183–190 (1988)
30. Zhou, L.: Strictly fair allocations in large exchange economies. Journal of Economic Theory **57**(1), 158 – 175 (1992)