Achieving Envy-Freeness with Limited Subsidies under Dichotomous Valuations

Siddharth Barman[1], **Anand Krishna**[1], **Yadati Narahari**[1] and **Soumyarup Sadhukan**[1,2]
[1]Indian Institute of Science, Bengaluru
[2]Indian Institute of Technology, Kanpur
{barman, anandkrishna, narahari, soumyarups}@iisc.ac.in

Abstract

We study the problem of allocating indivisible goods among agents in a fair manner. While envy-free allocations of indivisible goods are not guaranteed to exist, envy-freeness can be achieved by additionally providing some subsidy to the agents. These subsidies can be alternatively viewed as a divisible good (money) that is fractionally assigned among the agents to realize an envy-free outcome. In this setup, we bound the subsidy required to attain envy-freeness among agents with dichotomous valuations, i.e., among agents whose marginal value for any good is either zero or one.

We prove that, under dichotomous valuations, there exists an allocation that achieves envy-freeness with a per-agent subsidy of either 0 or 1. Furthermore, such an envy-free solution can be computed efficiently in the standard value-oracle model. Notably, our results hold for general dichotomous valuations and, in particular, do not require the (dichotomous) valuations to be additive, submodular, or even subadditive. Also, our subsidy bounds are tight and provide a linear (in the number of agents) factor improvement over the bounds known for general monotone valuations.

1 Introduction

Discrete fair division is an extremely active field of work at the interface of mathematical economics and computer science [Brandt *et al.*, 2016; Endriss, 2017]. This field addresses the problem of finding fair allocations of indivisible goods (i.e., resources that cannot be fractionally assigned) among agents with individual preferences. Solution concepts and algorithms developed here address several real-world settings, such as assignment of housing units [Benabbou *et al.*, 2020; Deng *et al.*, 2013], course allocation [Budish *et al.*, 2017], and inheritance division; the widely-used platform spliddit.org [Goldman and Procaccia, 2015] provides fair-division methods for a range of other allocation problems.

The quintessential and classic [Foley, 1966; Varian, 1974] notion of fairness in mathematical economics is that of envy-freeness, which requires that each agent weakly prefers the bundle assigned to her over that of any other agent. However, in the case of indivisible goods, an envy-free allocation is not guaranteed to exist; consider the (ad nauseam) example of two agents and one good.

Motivated, in part, by such considerations, a notable thread of research in discrete fair division aims to achieve the ideal of envy-freeness with the use of subsidies [Halpern and Shah, 2019; Brustle *et al.*, 2020; Goko *et al.*, 2021; Caragiannis and Ioannidis, 2021]. The objective is to provide each agent i—in addition to a bundle—an appropriate subsidy p_i such that envy-freeness is achieved overall. That is, the aim is to identify an allocation of the goods and subsidies such that each agent i's value for her bundle plus her subsidy, p_i, is at least as large as i's value for any other agent j's bundle plus the subsidy p_j.

The subsidy model can be alternatively viewed as a setting wherein we have a divisible good (money), along with the indivisible ones. The divisible good can be fractionally assigned among the agents towards achieving fair (envy-free) outcomes. A natural question in this context is to understand how much subsidy is required to eliminate all envy. This question was in fact addressed in the classic work of Maskin [1987] for the case of unit-demand agents. Considering a scaling, wherein each agent has value at most 1 for any good, Maskin showed that a total subsidy of $(n-1)$ suffices;[1] throughout, n denotes the number of agents participating in the fair division exercise. This unit-demand setting is referred to as the rent division problem and has been studied over the past few decades; see, e.g., [Aziz, 2020] and references therein. Going beyond rent division, the above-mentioned results address settings in which agents' preferences span the subsets of the goods. In particular, the work of Halpern and Shah [2019] shows that, if the n agents have binary additive valuations, then envy-freeness can be achieved with a total subsidy of $(n-1)$. A similar result was obtained for binary submodular valuations by Goko et al. [2021].

The current work contributes to this line of work with a focus on valuations that have binary marginals. Specifically, we address fair division instances in which, for each agent i, the marginal value of any good g relative to any subset S is either zero or one, $v_i(S \cup \{g\}) - v_i(S) \in \{0,1\}$; here, set

[1] Here, a valuation scaling is unavoidable, since the obtained guarantee is an absolute bound (on the total subsidy).

function v_i denotes the valuation of agent i. Such set functions are referred to as dichotomous valuations and have received significant attention in various fair division settings, e.g., [Bogomolnaia *et al.*, 2005; Bouveret and Lang, 2008; Kurokawa *et al.*, 2018; Babaioff *et al.*, 2021]. Indeed, dichotomous valuations—and restricted subclasses thereof—model preferences in several application domains; see, e.g., [Roth *et al.*, 2005; Deng *et al.*, 2013; Benabbou *et al.*, 2020].

The following stylized example illustrates the applicability of dichotomous valuations and the use of subsidies: consider a scheduling setting in which time-slots (indivisible goods) have be to assigned among employees (agents) whose nonzero marginal values for the slots are the same; equivalently, the agents have binary marginals for the slots. An employee's preference on whether a time-slot can be utilized (i.e., the slot has a nonzero marginal value) could depend on multiple factors, such as contiguity of slots, shift rotations, and time of day. Still, given that the current work holds for general dichotomous valuations,[2] finding a fair allocation in such a scheduling setting falls under the purview of the current work. Also, in this context, subsidies can be viewed as bonuses paid to the employees.

Our Results. We prove that, under dichotomous valuations, there exists an allocation that achieves envy-freeness with a per-agent subsidy of either 0 or 1 (Theorem 4). Furthermore, such an envy-free solution can be computed efficiently in the standard value-oracle model. Our per-agent subsidy bound implies that, for dichotomous valuations, a total subsidy of at most $(n-1)$ suffices to realize envy-freeness. Indeed, this guarantee is tight: consider an instance with n agents and a single unit-valued good. In this instance, to achieve envy-freeness under any allocation, $(n-1)$ agents would each require a subsidy of 1.

Additional Related Work. As mentioned previously, Halpern and Shah [2019] showed that if the agents' valuations are binary additive, then envy-freeness can be achieved with a cumulative subsidy of $(n-1)$. The work of Goko et al. [2021] provides an analogous result for binary submodular valuations and further develops a strategy-proof mechanism for this setting. Note that binary additive and binary submodular functions constitute subclasses of dichotomous valuations.

Building upon [Halpern and Shah, 2019], Brustle et al. [2020] proved that a subsidy of at most 1 per agent suffices for additive valuations; this result is established under a standard scaling, wherein the value of any good across all agents is at most 1. Brustle et al. [2020] also obtained an $O(n^2)$ upper bound on the total required subsidy for general monotone valuations. The current work achieves a factor n improvement over this upper bound for dichotomous valuations in particular.

Caragiannis and Ioannidis [2021] study the problem of finding allocations that realize envy-freeness with as little subsidy as possible. Since this problem is NP-hard, they focus on approximation algorithms and, in particular, develop a fully polynomial-time approximation scheme for instances with a constant number of agents. Narayan et al. [2021] show that one can achieve envy-freeness with transfers, while incurring a constant-factor loss in Nash social welfare.

We note that the framework of subsidies complements discrete fair division results which strive towards existential guarantees by relaxing the envy-freeness criterion. Here, a well-studied solution concept (relaxation) is envy-freeness up to one good (EF1) [Budish, 2011]: an allocation is said to be EF1 iff each agent values her bundle at least as much as any other agent's bundle, up to the removal of some good from the other agent's bundle. EF1 allocations are known to exist under monotone valuations [Lipton *et al.*, 2004].

2 Notation and Preliminaries

We study the problem of allocating m indivisible goods among n agents, with subsidies, in a fair manner. The cardinal preference of each agent $i \in [n]$, over the subsets of goods, is specified via valuation $v_i : 2^{[m]} \to \mathbb{R}_+$. Here, $v_i(S) \in \mathbb{R}_+$ denotes the valuation that agent i has for a subset of goods $S \subseteq [m]$. We will throughout represent a discrete fair division instance via the tuple $\langle [n], [m], \{v_i\}_{i=1}^n \rangle$. Our algorithms work in the standard value-oracle model. That is, for each valuation v_i, we only require an oracle that—when queried with a subset $S \subseteq [m]$—provides the value $v_i(S)$.

This work focuses on valuations that have binary marginals, i.e., for each agent $i \in [n]$, the marginal value of including any good $g \in [m]$ in any subset $S \subseteq [m]$ is either zero or one: $v_i(S \cup \{g\}) - v_i(S) \in \{0, 1\}$. We will refer to such set functions v_is as dichotomous valuations. Note that a dichotomous valuation v_i is monotone: $v_i(A) \leq v_i(B)$ for all subsets $A \subseteq B \subseteq [m]$. Also, we will assume that the agents' valuations satisfy $v_i(\emptyset) = 0$.

An allocation $\mathcal{A} = (A_1, A_2, \ldots, A_n)$ is an ordered collection of n pairwise disjoint subsets, $A_1, A_2, \ldots, A_n \subseteq [m]$, wherein the subset of goods A_i is assigned to agent $i \in [n]$. We refer to each such subset A_i as a bundle. Note that an allocation can be partial in the sense that $\cup_{i=1}^n A_i \neq [m]$. For disambiguation, we will use the term complete allocation to denote allocations wherein all the goods have been assigned and, otherwise, use the term partial allocation. The social (utilitarian) welfare of an allocation $\mathcal{A} = (A_1, \ldots, A_n)$ is the sum of the values that it generates among the agents, $\sum_{i=1}^n v_i(A_i)$.

The quintessential notion of fairness is that of envy-freeness [Foley, 1966]. This notion is defined next for allocations of indivisible goods.

Definition 1. *An allocation $\mathcal{A} = (A_1, \ldots, A_n)$ is said to be envy-free (*EF*) iff $v_i(A_i) \geq v_i(A_j)$ for all agents $i, j \in [n]$.*

Since, in the context of indivisible goods, envy-free allocations are not guaranteed to exist, a realizable desideratum is to achieve envy-freeness with the use of subsidies. The objective here is to provide each agent $i \in [n]$—in addition to a bundle A_i—an appropriate subsidy $p_i \geq 0$ such that envy-freeness is achieved overall. We will write vector $p = (p_1, p_2, \ldots, p_n) \in \mathbb{R}_+^n$ to denote the subsidies assigned

[2] In particular, we do not require the dichotomous valuations to be additive, submodular, or even subadditive.

to the agents. Our overarching aim is to find a complete allocation $\mathcal{A} = (A_1, \ldots, A_n)$ and a (bounded) subsidy vector $p = (p_1, \ldots, p_n)$ that ensure envy-freeness for each agent i, i.e., agent i's value for her bundle (A_i) plus her subsidy, p_i, is at least as large as i's value for any other agent j's bundle (A_j) plus the subsidy p_j. Formally,

Definition 2 (Envy-Free Solution)**.** *An allocation $\mathcal{A} = (A_1, \ldots, A_n)$ and a subsidy vector $p = (p_1, \ldots, p_n)$ are said to constitute an envy-free solution, $(\mathcal{A}, p)$, iff $v_i(A_i) + p_i \geq v_i(A_j) + p_j$ for all agents $i, j \in [n]$.*

The term *envy-freeable* will be used to denote any allocation $\mathcal{A}$ for which there exists a subsidy vector p such that $(\mathcal{A}, p)$ is an envy-free solution. Also, we will throughout write $\mathrm{M}(p)$ to denote the set of agents that receive maximum subsidy under $p = (p_1, \ldots, p_n)$, i.e., $\mathrm{M}(p) := \{i \in [n] \mid p_i \geq p_j \text{ for all } j \in [n]\}$.

2.1 Characterizing Envy-Freeable Allocations

The work of Halpern and Shah [2019] provides a useful characterization of envy-freeable allocations.[3] Using the characterization, one can determine (in polynomial time) whether or not a given allocation $\mathcal{B}$ is envy-freeable, i.e., whether $\mathcal{B}$ can be coupled with a subsidy vector q to realize an envy-free solution $(\mathcal{B}, q)$. Furthermore, for any envy-freeable allocation $\mathcal{B}$, the characterization leads to a polynomial-time algorithm to compute a corresponding subsidy vector q.

The characterization is obtained by considering a directed graph that captures the envy among the agents. Specifically, for an allocation $\mathcal{A} = (A_1, \ldots, A_n)$, the *envy-graph $G_{\mathcal{A}}$* is a complete, weighted, directed graph wherein the vertices correspond to the agents and each directed edge (i, j) has weight $w_{\mathcal{A}}(i, j) := v_i(A_j) - v_i(A_i)$. Note that the weight of the edge (i, j) is the envy that agent i has towards j, and the edge weights can be negative. The weight of any (directed) path P in the graph $G_{\mathcal{A}}$ will be denoted by $w_{\mathcal{A}}(P)$; in particular, $w_{\mathcal{A}}(P)$ is the sum of the weights of the edges along the path, $w_{\mathcal{A}}(P) = \sum_{(i,j) \in P} w_{\mathcal{A}}(i, j)$.

To achieve envy-freeness in certain settings, we will have to reassign the bundles among the agents. In particular, for an allocation $\mathcal{A} = (A_1, \ldots, A_n)$ and permutation $\sigma \in \mathbb{S}_n$ (over the n agents), write $\mathcal{A}_\sigma := (A_{\sigma(1)}, A_{\sigma(2)}, \ldots, A_{\sigma(n)})$ to denote the allocation wherein each agent $i \in [n]$ receives the bundle $A_{\sigma(i)}$.

The theorem below states the characterization of envy-freeable allocations.

Theorem 1. *[Halpern and Shah, 2019] For any allocation $\mathcal{A} = (A_1, \ldots, A_n)$, the following statements are equivalent:*

(i) $\mathcal{A}$ is envy-freeable.

(ii) $\mathcal{A}$ maximizes the social welfare across all reassignments of its bundles among the agents. That is, for every permutation σ over $[n]$, we have $\sum_{i=1}^{n} v_i(A_i) \geq \sum_{i=1}^{n} v_i(A_{\sigma(i)})$.

(iii) The envy-graph $G_{\mathcal{A}}$ has no positive-weight cycles.

Condition (ii) of Theorem 1 implies that, starting with an arbitrary allocation $\mathcal{B} = (B_1, \ldots, B_n)$, we can always find an envy-freeable allocation by reassigning the bundles B_is among the agents. In particular, if σ is any maximum-weight matching between the agents and the bundles, then B_σ is an envy-freeable allocation. Note that such a matching can be computed in polynomial-time.

Complementing the characterization, Theorem 2 (below) identifies an efficient algorithm to compute a subsidy vector for any given envy-freeable allocation $\mathcal{A}$. Here, $\ell_{\mathcal{A}}(i)$ is used to denote the maximum weight of any path (in $G_{\mathcal{A}}$) which starts at $i \in [n]$. Since the envy-graph $G_{\mathcal{A}}$ (for an envy-freeable allocation $\mathcal{A}$) does not contain any positive-weight cycle (condition (iii) in Theorem 1), the weight $\ell_{\mathcal{A}}(i)$ is well-defined and can be computed efficiently. In particular, we can apply the Floyd-Warshall algorithm, after negating the edge weights. The theorem asserts that providing a subsidy of $\ell_{\mathcal{A}}(i)$, to each agent i, suffices to achieve envy-freeness.

Theorem 2. *[Halpern and Shah, 2019] For any envy-freeable allocation $\mathcal{A}$, a subsidy of $p_i^* := \ell_{\mathcal{A}}(i)$, for all agents $i \in [n]$, realizes an envy-free solution $(\mathcal{A}, p^*)$; here subsidy vector $p^* = (p_1^*, \ldots, p_n^*)$. Furthermore, for any other vector $p = (p_1, \ldots, p_n) \in \mathbb{R}_+^n$, such that $(\mathcal{A}, p)$ is envy-free, we have $p_i^* \leq p_i$ for all $i \in [n]$.*

For any subsidy vector $p = (p_1, \ldots, p_n)$ and permutation σ over $[n]$, let $p_\sigma = (p_{\sigma(1)}, p_{\sigma(2)}, \ldots, p_{\sigma(n)})$ denote the subsidies obtained by reassigning according to σ, i.e., under p_σ, agent i receives a subsidy $p_{\sigma(i)}$.

The following lemma shows that if reassigning bundles, according to some permutation σ, maintains envy-freeability, then, in fact, reassigning the subsidies—also according to σ—preserves envy-freeness. A proof of this lemma appears in the full version of this work [Barman *et al.*, 2022].

Lemma 3. *Let $(\mathcal{A}, p)$ be an envy-free solution and let σ be a permutation such that allocation $\mathcal{B} := \mathcal{A}_\sigma$ is envy-freeable. Then, $(\mathcal{B}, p_\sigma)$ is an envy-free solution as well.*

3 Our Results

The next theorem is the main result of the current work and it shows that in any discrete fair division instance with dichotomous valuations there exists an envy-free solution with a subsidy of at most 1 per agent.

Theorem 4. *For any discrete fair division instance $\langle [n], [m], \{v_i\}_{i=1}^n \rangle$ with dichotomous valuations, there exists an envy-free solution $(\mathcal{A}, p)$ such that $p \in \{0, 1\}^n$. Furthermore, given value oracle access to the v_is, such an envy-free solution, $(\mathcal{A}, p)$, can be computed in polynomial time.*

This theorem implies that, to achieve envy-freeness under dichotomous valuations, the total required subsidy is at most $(n - 1)$, i.e., $\sum_{i=1}^{n} p_i \leq n - 1$. Note that the case in which $p_i = 1$, for all agents i, can be addressed by giving each agent a subsidy of 0.

4 Finding Envy-Free Solutions for Dichotomous Valuations

For dichotomous valuations, our algorithm ALG (Algorithm 1) finds an envy-free solution $(\mathcal{A}, p)$ such that the sub-

[3] An analogous result was developed by Aragones [1995] in the context of fair rent division.

sidy vector $p \in \{0,1\}^n$. In particular, with iteration count t, the algorithm inductively maintains an envy-free solution $(\mathcal{A}^t, p^t)$ wherein $\mathcal{A}^t = (A_1^t, \ldots A_n^t)$ is a (partial) envy-freeable allocation and the subsidy vector $p^t \in \{0,1\}^n$. In every iteration t, the algorithm selects an unallocated good $g \in [m] \setminus (\cup_{i=1}^n A_i^t)$, includes it in one of the current bundles, $A_1^t, \ldots, A_n^t$, and possibly reassigns the bundles among the agents such that the updated allocation, $\mathcal{A}^{t+1}$, is also envy-freeable with subsidies in $\{0,1\}$.

We note that a relatively simple case occurs if—in any iteration t and for an unallocated good g—there exists an agent $k \in [n]$ such that (a) the marginal value of good g for agent k is equal to 1 (i.e., $v_k(A_k^t \cup \{g\}) - v_k(A_k^t) = 1$) and (b) agent k's subsidy (under $p^t = (p_1^t, \ldots, p_n^t)$) is at least as large as that of any other agent ($p_k^t \geq p_j^t$ for all j). In such a case, the allocation $\mathcal{A}^{t+1}$ obtained by assigning the good g to the agent k is envy-freeable, since the social welfare of $\mathcal{A}^{t+1}$ is one more than the social welfare of $\mathcal{A}^t$ and this is the maximum possible among all the reassignments of the bundles in $\mathcal{A}^{t+1}$ (see condition (ii) in Theorem 1). Furthermore, for the allocation $\mathcal{A}^{t+1}$, the accompanying subsidies remain in $\{0,1\}$. This essentially follows from the facts that k was one of the most subsidized agents and, when we assign the good g to agent k, only the weights of the edges incident on k (in the updated envy graph) change. In particular, the weight of the edges going out of k necessarily decrease by 1 and for each incoming edge the weight increases by at most 1.[4] Using these observations one can show that the weights of all paths remain below 1 and, hence (via Theorem 2), the subsidies remain in $\{0,1\}$.

Even if there does not exist an agent k that directly satisfies the above-mentioned conditions (a) and (b), we can still try to first reassign the current bundles (among the agents) and then look for such an agent. Specifically, the case detailed above continues to be relevant if there exists a permutation σ such that allocation $\mathcal{B} = \mathcal{A}_\sigma^t$ is envy-freeable,[5] and, under $\mathcal{B}$, there exists an agent κ that satisfies conditions (a) and (b). Encapsulating this case,[6] we will say that the current allocation $\mathcal{A}^t$ is *extendable* with good g, iff the desired permutation σ and agent κ exist; see Definition 3.

Section 4.1 develops a subroutine, EXTEND (Algorithm 2), that efficiently identifies whether the current allocation $\mathcal{A}^t$ is extendable. If the allocation is extendable, then the subroutine returns the relevant permutation σ and the agent κ. Using this subroutine, our algorithm ALG addresses the case of extendable solutions (in Lines 4 to 6). In particular, ALG allocates good g and updates the envy-free solution $(\mathcal{A}^t, p^t)$ to $(\mathcal{A}^{t+1}, p^{t+1})$, while maintaining the invariant that the subsidies are either 0 or 1.

In the complementary case, wherein the current allocation $\mathcal{A}^t$ is not extendable, ALG (in Line 8) invokes the subroutine FINDSINK (detailed in Section 4.2). A key technical insight here is that, under dichotomous valuations and for an allocation $\mathcal{A}^t$ that is *not* extendable with a good g, there necessarily exists an agent s such that assigning g to s maintains the subsidies in $\{0,1\}$. The subroutine FINDSINK directly finds such an agent by trying to assign g to different agents iteratively and checking whether subsidies remain in $\{0,1\}$. It is relevant to note that, while the subroutine is simple in design, its analysis requires intricate existential arguments. In particular, Section 4.2 establishes that FINDSINK finds such an agent $s \in [n]$ in polynomial time and, hence, provides a constructive proof of existence of the desired agent s.

Algorithm 1 ALG

```
Input: Instance $\langle [n], [m], \{v_i\}_{i \in [n]} \rangle$ with value oracle access
       to dichotomous valuations $v_i$s.
Output: Allocation $\mathcal{A}$ and subsidy vector $p$.

 1: Initialize index $t = 1$ along with bundles $A_1^t = \ldots = A_n^t = \emptyset$ and subsidies
    $p_1^t = \ldots = p_n^t = 0$. {Here, allocation $\mathcal{A}^t = (\emptyset, \ldots, \emptyset)$ and subsidy vector $p^t = (0, \ldots, 0)$.}
 2: while $[m] \setminus \left(\cup_{i \in [n]} A_i^t\right) \neq \emptyset$ do
 3:     Select any unallocated good $g \in [m] \setminus \left(\cup_{i \in [n]} A_i^t\right)$.
 4:     if $(\mathcal{A}^t, p^t)$ is extendable with good $g$ then
 5:         Set $(\sigma, \kappa) := \text{EXTEND}(\mathcal{A}^t, p^t, g)$ and set allocation $(B_1, \ldots, B_n) := \mathcal{A}_\sigma^t$.
            {$(B_1, \ldots, B_n)$ is obtained by reassigning bundles in $\mathcal{A}^t$, according to permutation $\sigma$.}
 6:         Set allocation $\mathcal{A}^{t+1} := (B_1, \ldots, B_\kappa \cup \{g\}, \ldots, B_n)$.
            {Good $g$ is included in agent $\kappa$'s bundle $B_\kappa$.}
 7:     else
 8:         Let agent $s := \text{FINDSINK}(\mathcal{A}^t, p^t, g)$.
 9:         Set allocation $\mathcal{A}^{t+1} := (A_1^t, \ldots, A_s^t \cup \{g\}, \ldots, A_n^t)$.
            {Good $g$ is included in agent $s$'s bundle $A_s^t$.}
10:     end if
11:     Compute subsidy vector $p^{t+1}$ for the allocation $\mathcal{A}^{t+1}$.
        {We will prove that, in all cases, $\mathcal{A}^{t+1}$ is envy-freeable and invoke Theorem 2 to compute the subsidies.}
12:     Update $t \leftarrow t + 1$.
13: end while
14: return allocation $\mathcal{A}^t = (A_1^t, \ldots, A_n^t)$ and subsidy vector $p^t = (p_1^t, \ldots, p_n^t)$.
```

Therefore, in every possible case, our algorithm ALG assigns a good and updates the allocation, all the while maintaining the invariant that the allocation in hand is envy-freeable and requires subsidies that are either 0 or 1.

For the analysis of the algorithm, we will use the following two propositions, which hold for dichotomous valuations. The proofs of the propositions appear in the full version of this paper [Barman *et al.*, 2022].

Proposition 5. *Let $\mathcal{Y} = (Y_1, \ldots, Y_n)$ be an envy-freeable (partial) allocation. Also, assume that, for an agent $x \in [n]$ and an unallocated good g, we have $v_x(Y_x \cup \{g\}) - v_x(Y_x) = 1$. Then, the allocation $(Y_1, \ldots, Y_x \cup \{g\}, \ldots, Y_n)$ is envy-freeable as well.*

[4] Note that under dichotomous valuations the edge weights in the envy graph are integers.

[5] Here, the envy-freeability of $\mathcal{B}$ ensures that the accompanying subsidies are still either 0 or 1; see Lemma 3.

[6] Note that with σ as the identity permutation, one can capture the setting wherein a reassignment of the bundles is not required.

Proposition 6. *For any envy-freeable allocation $\mathcal{Y} = (Y_1, \dots, Y_n)$, any agent x, and any (unallocated) good g, let allocation $\mathcal{Z} := (Y_1, \dots, Y_x \cup \{g\}, \dots, Y_n)$. Then, in the envy graph $G_\mathcal{Z}$, the weights of all the edges—except the ones incident on x—are the same as in $G_\mathcal{Y}$, i.e., for all edges (i, j), with $i, j \in [n] \setminus \{x\}$, we have $w_\mathcal{Z}(i, j) = w_\mathcal{Y}(i, j)$. Furthermore, the weights of edges (x, j) going out of x satisfy $w_\mathcal{Z}(x, j) \leq w_\mathcal{Y}(x, j)$, and the weights of edges (i, x) coming into x satisfy $w_\mathcal{Z}(i, x) \leq w_\mathcal{Y}(i, x) + 1$.*

For allocation $\mathcal{Z} := (Y_1, \dots, Y_x \cup \{g\}, \dots, Y_n)$, Proposition 6 implies that the weight of any path[7] P in the envy graph $G_\mathcal{Z}$ is at most one more than the weight of P in the envy graph $G_\mathcal{Y}$, i.e., $w_\mathcal{Z}(P) \leq w_\mathcal{Y}(P) + 1$.

4.1 Extendable Solutions

As mentioned previously, a relevant case for our algorithm occurs when, for an unallocated good g, there exists an agent κ such that (a) the marginal value of good g for agent κ is equal to one and (b) agent κ is one of the most subsidized agents. The notion of extendability (defined below) generalizes this case. Recall that $\mathrm{M}(p)$—for any subsidy vector $p = (p_1, \dots, p_n)$—denotes the set of the most subsidized agents, $\mathrm{M}(p) := \{i \in [n] \mid p_i \geq p_j \text{ for all } j \in [n]\}$.

Definition 3 (Extendable Solutions)**.** *An envy-free solution $(\mathcal{A}, p)$ is said to be extendable with good $g \in [m] \setminus \left(\cup_{i \in [n]} A_i\right)$ iff there exists a permutation σ (over $[n]$) such that:*

(i) Allocation $\mathcal{B} := \mathcal{A}_\sigma$ and subsidy vector $q := p_\sigma$ constitute an envy-free solution $(\mathcal{B}, q)$, and

(ii) There exists an agent $\kappa \in \mathrm{M}(q)$ with the property that $v_\kappa(B_\kappa \cup \{g\}) - v_\kappa(B_\kappa) = 1$.

The following lemma shows that, for an extendable solution $(\mathcal{A}, p)$, we can allocate the good g and still maintain per-agent subsidy to be either 0 or 1.

Lemma 7. *In a fair division instance with dichotomous valuations, let $(\mathcal{A}, p)$ be an envy-free solution extendable with good g and permutation σ. Also, assume that the subsidies $p \in \{0, 1\}^n$. Then, the allocation $(B_1, \dots, B_\kappa \cup \{g\}, \dots, B_n)$ is also envy-freeable with a subsidy of either 0 or 1 for each agent. Here, allocation $(B_1, \dots, B_n) := \mathcal{A}_\sigma$ and κ is the agent identified in the extendability criteria (Definition 3).*

The proof of Lemma 7 appears in the full version of this work [Barman *et al.*, 2022].

The following lemma shows that the subroutine EXTEND efficiently identifies whether a given solution $(\mathcal{A}, p)$ is extendable, i.e., the subroutine efficiently tests whether Definition 3 holds, or not. See the full version of this paper for the proof of the lemma [Barman *et al.*, 2022].

Lemma 8. *In a fair division instance with dichotomous valuations, let $(\mathcal{A}, p)$ be an envy-free solution.*

- *If $(\mathcal{A}, p)$ is extendable with good $g \in [m] \setminus (\cup_{i=1}^n A_i)$, then the* EXTEND *subroutine returns a permutation σ and an agent κ that satisfy the extendability criteria (Definition 3).*
- *Otherwise, if $(\mathcal{A}, p)$ is not extendable with good g, then subroutine* EXTEND *correctly reports as such.*

Algorithm 2 EXTEND

```
Input: Instance $\langle [n], [m], \{v_i\}_{i \in [n]} \rangle$, an envy-free solution $(\mathcal{A}, p)$, and a good $g \in [m]$.
Output: A permutation and an agent that extend the input $(\mathcal{A}, p)$ with good $g$, or return that it is not extendable.
for each $k \in [n]$ and $\ell \in \mathrm{M}(p)$ with the property that $v_k(A_\ell \cup \{g\}) - v_k(A_\ell) = 1$ do
    Consider a complete bipartite graph $H$ between sets $[n] \setminus \{k\}$ and $[n] \setminus \{\ell\}$. For each edge $(i, j)$ in $H$ set the weight to be $v_i(A_j)$.
    Compute a maximum-weight matching $\rho$ in $H$.
    Set $\rho(k) = \ell$. {$\rho$ is a permutation over $[n]$.}
    If $\sum_{i=1}^n v_i(A_{\rho(i)}) \geq \sum_{i=1}^n v_i(A_i)$, then return $(\rho, k)$.
end for
return "$(\mathcal{A}, p)$ is not extendable with good $g$."
```

4.2 Non-Extendable Solutions

This section addresses the case wherein the maintained allocation is not extendable. The subroutine FINDSINK takes as input an envy-free solution $(\mathcal{A}, p)$—with subsidy vector $p \in \{0, 1\}^n$—and a good g. The subroutine tentatively assigns the good to one agent at a time and checks if the subsidies remain in $\{0, 1\}$. We will show that if the input $(\mathcal{A}, p)$ is not extendable, then FINDSINK necessarily succeeds in finding such an agent (Lemmas 10 and 11).

Algorithm 3 FINDSINK

```
Input: Instance $\langle [n], [m], \{v_i\}_{i \in [n]} \rangle$, an envy-free solution $(\mathcal{A}, p)$, and a good $g \in [m]$.
Output: An agent $s$.
Select an arbitrary agent $s \in \mathrm{M}(p)$ and set allocation $\mathcal{X} = (A_1, A_2, \dots, A_s \cup \{g\}, \dots, A_n)$.
{We will show that—for non-extendable inputs—allocation $\mathcal{X}$ is envy-freeable.}
Compute the subsidy vector $(\varphi_1, \dots, \varphi_n)$ for allocation $\mathcal{X}$. {See Theorem 2.}
while there exists an agent $j$ with subsidy $\varphi_j \geq 2$ do
    Update $s \leftarrow j$ and update $\mathcal{X} = (A_1, A_2, \dots, A_s \cup \{g\}, \dots, A_n)$.
    Compute the subsidy vector $(\varphi_1, \dots, \varphi_n)$ for allocation $\mathcal{X}$.
end while
return agent $s$.
```

The lemma below shows that all the agents s considered by FINDSINK are among the most subsidized ones, under p, and the considered allocations $\mathcal{X} := (A_1, \dots, A_s \cup \{g\}, \dots, A_n)$ are envy-freeable. Note that, throughout the execution of the subroutine, $(\mathcal{A}, p)$ remains unchanged.

Lemma 9. *In a fair division instance with dichotomous valuations, let $(\mathcal{A}, p)$ be an envy-free solution with subsidies $p \in \{0, 1\}^n$. If input $(\mathcal{A}, p)$ is not extendable with the given*

[7] Note that, by definition, a path joins *distinct* vertices.

good g, then each agent s considered in FINDSINK *(Line 4) belongs to the set* $\mathrm{M}(p)$ *and each considered allocation* $\mathcal{X} := (A_1, \ldots, A_s \cup \{g\}, \ldots, A_n)$ *is envy-freeable.*

The proof of the Lemma 9 appears in the full version [Barman *et al.*, 2022].

The next lemma is a key technical result for FINDSINK. It shows that the subroutine terminates in polynomial time, when the input $(\mathcal{A}, p)$ is not extendable. The proof of this lemma entails intricate existential arguments; we defer the proof to the full version of this paper [Barman *et al.*, 2022].

Lemma 10. *In a fair division instance with dichotomous valuations, let $(\mathcal{A}, p)$ be an envy-free solution with subsidies $p \in \{0,1\}^n$. Also, assume that $(\mathcal{A}, p)$ is not extendable with good g. Then, given $(\mathcal{A}, p)$ and g as input, the subroutine* FINDSINK *terminates in polynomial time.*

The subroutine FINDSINK is designed to terminate only when it has identified an agent s to whom we can assign the good g and maintain the subsidies to be in $\{0, 1\}$. The next lemma formalizes this observation.

Lemma 11. *In a fair division instance with dichotomous valuations, let $(\mathcal{A}, p)$ be an envy-free solution with $p \in \{0,1\}^n$. Also, assume that $(\mathcal{A}, p)$ is not extendable with good g and let $s \in [n]$ be the agent returned by* FINDSINK *(given $(\mathcal{A}, p)$ and g as input). Then, the allocation $(A_1, \ldots, A_s \cup \{g\}, \ldots, A_n)$ is envy-freeable with a subsidy of either* 0 *or* 1 *for each agent.*

Proof. Since s is one of the agents selected in FINDSINK, Lemma 9 implies that the allocation $(A_1, \ldots, A_s \cup \{g\}, \ldots, A_n)$ is envy-freeable. Furthermore, the termination condition of the while-loop in FINDSINK (Line 3) ensures that, for the allocation $(A_1, \ldots, A_s \cup \{g\}, \ldots, A_n)$, the required subsidies are less than 2. Since the subsidies are non-negative integers (for dichotomous valuations), they are either 0 or 1. This completes the proof. □

4.3 Proof of Theorem 4

In this section, we will prove Theorem 4 by establishing that the solution returned by ALG (Algorithm 1) is envy-free and requires, for each agent, a subsidy of either 0 or 1.

Theorem 4. *For any discrete fair division instance $\langle [n], [m], \{v_i\}_{i=1}^n \rangle$ with dichotomous valuations, there exists an envy-free solution $(\mathcal{A}, p)$ such that $p \in \{0,1\}^n$. Furthermore, given value oracle access to the v_is, such an envy-free solution, $(\mathcal{A}, p)$, can be computed in polynomial time.*

Proof. We will show that, with iteration counter $t \geq 1$, the algorithm inductively maintains an envy-free solution $(\mathcal{A}^t, p^t)$ wherein $\mathcal{A}^t = (A_1^t, \ldots A_n^t)$ is a (partial) envy-freeable allocation and the subsidy vector $p^t \in \{0,1\}^n$. The initialization in Line 1 of ALG ensures that this property holds for $t = 1$, since the empty allocation $\mathcal{A}^1$ is envy-freeable with zero subsidies.

Next, consider any iteration $t \geq 1$ of the while-loop (Line 2) in ALG and let g be the good selected in Line 3. For the current solution $(\mathcal{A}^t, p^t)$, there are two complementary and exhaustive cases:

Case I*: $(\mathcal{A}^t, p^t)$ is extendable with good g.* Here, Lemma 8 implies that the permutation σ and the agent κ returned by the EXTEND subroutine satisfy the extendability criteria. Hence, by Lemma 7, we get that assigning g to agent κ's bundle—as in Lines 5 and 6—leads to an envy-free solution $(\mathcal{A}^{t+1}, p^{t+1})$ with $p^{t+1} \in \{0,1\}^n$.

Case II*: $(\mathcal{A}^t, p^t)$ is not extendable with good g.* This case will be correctly identified in ALG (Lemma 8) and the algorithm will then invoke the FINDSINK subroutine (Line 8). The subroutine in turn will find (in polynomial time) an agent $s \in [n]$ (Lemma 10) such that assigning the good g to agent s—as in Line 9—provides an envy-free solution $(\mathcal{A}^{t+1}, p^{t+1})$ with $p^{t+1} \in \{0,1\}^n$ (Lemma 11).

Therefore, in both cases, the updated solution $(\mathcal{A}^{t+1}, p^{t+1})$ is envy-free with $p^{t+1} \in \{0,1\}^n$. This, overall, shows that the returned solution $(\mathcal{A}, p)$ satisfies the desired properties.

Furthermore, note that the subroutines EXTEND and FINDSINK run in polynomial time, and only require value queries. This establishes the efficiency of ALG and completes the proof. □

5 Conclusion and Future Work

We prove that, under dichotomous valuations, envy-freeness can always be achieved with a subsidy of at most 1 per agent. This bound is tight and our proof is constructive. Specifically, our algorithm assigns the goods iteratively while maintaining envy-freeness, with bounded subsidies.

Even though, at a high level, our algorithm might seem like a refinement of the method used for finding EF1 allocations [Lipton *et al.*, 2004], the two approaches are distinct. In particular, the current algorithm resolves positive-weight cycles (in the envy graph) and, by contrast, finding EF1 allocations (under monotone valuations) entails resolution of top trading cycles [Lipton *et al.*, 2004]. We can, in fact, construct an instance wherein a specific execution of the developed algorithm returns an allocation that is not EF1; see the full version for details [Barman *et al.*, 2022]. Hence, extending the current work to additionally obtain the EF1 guarantee is a relevant direction of future work. Another interesting direction would be to obtain tight subsidy bounds for general monotone valuations.

Acknowledgements

Siddharth Barman gratefully acknowledges the support of a Microsoft Research Lab (India) grant and an SERB Core research grant (CRG/2021/006165).

References

[Aragones, 1995] Enriqueta Aragones. A derivation of the money rawlsian solution. *Social Choice and Welfare*, 12(3):267–276, 1995.

[Aziz, 2020] Haris Aziz. Developments in multi-agent fair allocation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 13563–13568, 2020.

[Babaioff *et al.*, 2021] Moshe Babaioff, Tomer Ezra, and Uriel Feige. Fair and truthful mechanisms for dichotomous valuations. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35. AAAI, 2021.

[Barman *et al.*, 2022] Siddharth Barman, Anand Krishna, Y. Narahari, and Soumyarup Sadhukhan. Achieving envy-freeness with limited subsidies under dichotomous valuations. *ArXiv*, abs/2201.07419, 2022.

[Benabbou *et al.*, 2020] Nawal Benabbou, Mithun Chakraborty, Ayumi Igarashi, and Yair Zick. Finding fair and efficient allocations when valuations don't add up. In *International Symposium on Algorithmic Game Theory*, pages 32–46. Springer, 2020.

[Bogomolnaia *et al.*, 2005] Anna Bogomolnaia, Hervé Moulin, and Richard Stong. Collective choice under dichotomous preferences. *Journal of Economic Theory*, 122(2):165–184, 2005.

[Bouveret and Lang, 2008] Sylvain Bouveret and Jérôme Lang. Efficiency and envy-freeness in fair division of indivisible goods: Logical representation and complexity. *J. Artif. Int. Res.*, 32(1):525—-564, jun 2008.

[Brandt *et al.*, 2016] Felix Brandt, Vincent Conitzer, Ulle Endriss, Jérôme Lang, and Ariel D. Procaccia. *Handbook of Computational Social Choice*. Cambridge University Press, USA, 1st edition, 2016.

[Brustle *et al.*, 2020] Johannes Brustle, Jack Dippel, Vishnu V Narayan, Mashbat Suzuki, and Adrian Vetta. One dollar each eliminates envy. In *Proceedings of the 21st ACM Conference on Economics and Computation*, pages 23–39, 2020.

[Budish *et al.*, 2017] Eric Budish, Gérard P. Cachon, Judd B. Kessler, and Abraham Othman. Course match: A large-scale implementation of approximate competitive equilibrium from equal incomes for combinatorial allocation. *Oper. Res.*, 65:314–336, 2017.

[Budish, 2011] Eric Budish. The combinatorial assignment problem: Approximate competitive equilibrium from equal incomes. *Journal of Political Economy*, 119:1061–1103, 2011.

[Caragiannis and Ioannidis, 2021] Ioannis Caragiannis and Stavros Ioannidis. Computing envy-freeable allocations with limited subsidies. In *Workshop on Internet and Network Economics (WINE)*, 2021.

[Deng *et al.*, 2013] Yongheng Deng, Tien Foo Sing, and Chaoqun Ren. The story of singapore's public housing: From a nation of home-seekers to a nation of homeowners. In *The future of public housing*, pages 103–121. Springer, 2013.

[Endriss, 2017] Ulle Endriss. Trends in computational social choice. https://lulu.com, 2017. Accessed: 2022-06-11.

[Foley, 1966] Duncan Karl Foley. *Resource allocation and the public sector*. Yale University, 1966.

[Goko *et al.*, 2021] Hiromichi Goko, Ayumi Igarashi, Yasushi Kawase, Kazuhisa Makino, Hanna Sumita, Akihisa Tamura, Yu Yokoi, and M. Yokoo. Fair and truthful mechanism with limited subsidy. *ArXiv*, abs/2105.01801, 2021.

[Goldman and Procaccia, 2015] Jonathan Goldman and Ariel D. Procaccia. Spliddit: Unleashing fair division algorithms. *SIGecom Exch.*, 13(2):41—-46, Jan 2015.

[Halpern and Shah, 2019] Daniel Halpern and Nisarg Shah. Fair division with subsidy. In *International Symposium on Algorithmic Game Theory*, pages 374–389. Springer, 2019.

[Kurokawa *et al.*, 2018] David Kurokawa, Ariel D. Procaccia, and Nisarg Shah. Leximin allocations in the real world. *ACM Transactions on Economics and Computation (TEAC)*, 6:1–24, 2018.

[Lipton *et al.*, 2004] Richard J Lipton, Evangelos Markakis, Elchanan Mossel, and Amin Saberi. On approximately fair allocations of indivisible goods. In *Proceedings of the 5th ACM Conference on Electronic Commerce*, pages 125–131, 2004.

[Maskin, 1987] Eric S Maskin. On the fair allocation of indivisible goods. In *Arrow and the Foundations of the Theory of Economic Policy*, pages 341–349. Springer, 1987.

[Narayan *et al.*, 2021] Vishnu V. Narayan, Mashbat Suzuki, and Adrian Vetta. Two birds with one stone: Fairness and welfare via transfers. In *International Symposium on Algorithmic Game Theory*, pages 376–390. Springer, 2021.

[Roth *et al.*, 2005] Alvin E. Roth, Tayfun Sonmez, and M. Utku Unver. Pairwise kidney exchange. *Journal of Economic Theory*, 125(2):151–188, December 2005.

[Varian, 1974] Hal Varian. Equity, envy, and efficiency. *Journal of Economic Theory*, 9(1):63–91, 1974.