Greedy Algorithms for Maximizing Nash Social Welfare

Siddharth Barman
Indian Institute of Science
Bangalore, India
barman@iisc.ac.in

Sanath Kumar Krishnamurthy
Chennai Mathematical Institute
Kelambakkam, India
sanathkumar9@cmi.ac.in

Rohit Vaish
Indian Institute of Science
Bangalore, India
rohitv@iisc.ac.in

ABSTRACT

We study the problem of fairly allocating a set of indivisible goods among agents with additive valuations. The extent of fairness of an allocation is measured by its Nash social welfare, which is the geometric mean of the valuations of the agents for their bundles. While the problem of maximizing Nash social welfare is known to be APX-hard in general, we study the effectiveness of *simple*, *greedy* algorithms in solving this problem in two interesting special cases.

First, we show that a simple, greedy algorithm provides a 1.061-approximation guarantee when agents have *identical* valuations, even though the problem of maximizing Nash social welfare remains NP-hard for this setting. Second, we show that when agents have *binary* valuations over the goods, an *exact* solution (i.e., a Nash optimal allocation) can be found in polynomial time via a greedy algorithm. Our results in the binary setting extend to provide novel, exact algorithms for optimizing Nash social welfare under *concave* valuations. Notably, for the above mentioned scenarios, our techniques provide a *simple* alternative to several of the existing, more sophisticated techniques for this problem such as constructing equilibria of Fisher markets or using real stable polynomials.

KEYWORDS

Fair Division; Nash Social Welfare; Greedy Algorithms; Approximation Algorithms

ACM Reference Format:
Siddharth Barman, Sanath Kumar Krishnamurthy, and Rohit Vaish. 2018. Greedy Algorithms for Maximizing Nash Social Welfare. In *Proc. of the 17th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2018), Stockholm, Sweden, July 10–15, 2018,* IFAAMAS, 7 pages.

1 INTRODUCTION

We study the problem of fairly allocating a set of indivisible goods among agents with additive valuations for the goods. The fairness of an allocation is quantified by its *Nash social welfare* [11, 16], which is the geometric mean of the valuations of the agents under that allocation. The notion of Nash social welfare has traditionally been studied in the economics literature for *divisible* goods [15], where it is known to possess strong *fairness* and *efficiency* properties [20]. Besides, this notion is also attractive from a *computational* standpoint: For divisible goods, the Nash optimal allocation can be computed in polynomial time using the convex program of Eisenberg and Gale [9].

For *indivisible* goods, Nash social welfare once again provides notable fairness and efficiency guarantees [5]. However, the computational results in this setting are drastically different from its divisible counterpart. Indeed, it is known that the problem of maximizing Nash social welfare for indivisible goods is APX-hard when agents have additive valuations for the goods [12]. On the algorithmic side, the first constant-factor (specifically, 2.89) approximation for this problem was provided by Cole and Gkatzelis [7]. This approximation factor was subsequently improved to e [2], 2 [6] and, most recently, to 1.45 [3]. Similar approximation guarantees have also been developed for more general market models such as piecewise linear concave utilities [1], budget additive valuations [10], and multi-unit markets [4]. By and large, these approaches rely on either constructing an appropriate fractional equilibrium of a Fisher market and later rounding it to an integral allocation, or using real stable polynomials. Although these approaches offer strong approximation guarantees in very general market models, they are also (justifiably) more involved and often lack a *combinatorial* interpretation. Our interest in this work, therefore, is to understand the power of simple, combinatorial algorithms (in particular, *greedy* techniques) in solving interesting special cases of this problem.

Our results and techniques. We consider the Nash social welfare objective (NSW) as a measure of fairness in and of itself, and develop greedy algorithms for maximizing NSW either exactly or approximately. We focus on two special classes of additive valuations, namely *identical* valuations (i.e., for any good j and any pair of agents i, k, the value of the good j for i is equal to the value of the good for k; $v_{i,j} = v_{k,j}$) and *binary* valuations (i.e., for every agent i and good j, agent i's value for j is either 0 or 1; $v_{i,j} \in \{0, 1\}$). The class of identical valuations is well-studied in the approximation algorithms literature, and binary valuations capture the setting where each agent finds a good either acceptable or not.

For *identical* valuations, we show that a simple greedy algorithm provides a 1.061-approximation to the optimal Nash social welfare (Theorem 3.1). Note that the problem of maximizing Nash social welfare remains NP-hard even for identical valuations (via a reduction from the Partition problem [19]). Our algorithm (Algorithm 1) works by allocating the goods one by one in descending order of value. At each step, a good is allocated to the agent with the least valuation. This implicitly corresponds to greedily choosing an agent that provides the maximum improvement in NSW. We show that the allocation returned by our algorithm satisfies an approximate version of *envy-freeness* property (Lemma 4.1), and that any allocation with this property gives the desired approximation guarantee (Lemma 4.2). We remark that a polynomial time approximation scheme (PTAS) is already known for this problem [17]. However, this scheme uses Lenstra's algorithm for integer programs [14] as a subroutine, and hence is not combinatorial. Moreover, despite

Proc. of the 17th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2018), M. Dastani, G. Sukthankar, E. André, S. Koenig (eds.), July 10–15, 2018, Stockholm, Sweden.

being polynomial time in principle, the actual running time of such algorithms often scales rather poorly. By contrast, our algorithm involves a single sorting step and at most one $\min(\cdot)$ operation for each good, thus it requires $O(m \log m + mn)$ time overall.

For *binary* valuations, we show that an *exact* solution to the problem (i.e., a Nash optimal allocation) can be found by a greedy algorithm in polynomial time (Theorem 3.2). However, unlike the algorithm for identical valuations which greedily picks an *agent*, our algorithm for binary valuations makes a greedy decision with respect to *swaps* (or *chains* of swaps) between a *pair* of agents. A swap refers to taking a good away from one agent and giving it to another agent. A chain of swaps refers to a sequence of agents $u_1, u_2, \ldots, u_\ell$ and goods $j_1, j_2, \ldots, j_{\ell-1}$ such that j_1 is swapped from u_1 to u_2, j_2 is swapped from u_2 to u_3, and so on. Given any suboptimal allocation, our algorithm (Algorithm 2) checks for every pair of agents whether there exists a chain of swaps between them that improves NSW. The pair that provides the greatest improvement is chosen, and the corresponding swaps made. We show that the algorithm makes substantial progress toward the Nash optimal after each such reallocation (Lemma 5.1), which provides the desired running time and optimality guarantees. An interesting feature of our algorithm is that the guarantee for additive valuations extends to a more general utility model where the valuation of each agent is a *concave* function of its cardinality (i.e., the number of nonzero-valued goods in its bundle).[1] Prior work [8] has shown that a Nash optimal can be found efficiently under binary valuations (via reduction to minimum-cost flow problem). However, these techniques crucially rely on valuations depending *linearly* on cardinality, and it is unclear how to extend these to the aforementioned utility model. Our results, therefore, provide novel, exact algorithms for maximizing Nash social welfare under concave valuations.

2 PRELIMINARIES

Problem instance. An *instance* $\langle [n], [m], \mathcal{V} \rangle$ of the fair division problem is defined by (1) the set of $n \in \mathbb{N}$ *agents* $[n] = \{1, 2, \ldots, n\}$, (2) the set of $m \in \mathbb{N}$ *goods* $[m] = \{1, 2, \ldots, m\}$, and (3) the *valuation profile* $\mathcal{V} = \{v_1, v_2, \ldots, v_n\}$ that specifies the preferences of each agent $i \in [n]$ over the set of goods $[m]$ via a *valuation function* $v_i : 2^{[m]} \to \mathbb{Z}_+ \cup \{0\}$. Throughout, the valuations are assumed to be *additive*, i.e., for any agent $i \in [n]$ and any set of goods $G \subseteq [m]$, $v_i(G) \coloneqq \sum_{j \in G} v_i(\{j\})$, where $v_i(\{\emptyset\}) = 0$. We use the shorthand $v_{i,j}$ instead of $v_i(\{j\})$ for a singleton good $j \in [m]$. We will use Γ_i to denote the set of goods that are positively valued by agent i, i.e., $\Gamma_i \coloneqq \{j \in [m] : v_{i,j} > 0\}$.

Binary valuations. We say that agents have *binary* valuations if for each agent $i \in [n]$ and each good $j \in [m]$, $v_{i,j} \in \{0, 1\}$.

Identical valuations. We say that agents have *identical* valuations if for any good $j \in [m]$ and any pair of agents $i, k \in [n]$, we have $v_{i,j} = v_{k,j}$. For identical valuations, we will assume, without loss of generality, that the value of each good is nonzero. In addition, we will drop the agent-specific subscripts, and simply write $v(\{j\})$ to denote the value of good j.

[1]Notice that for binary and additive utilities, the valuation of each agent is a *linear* function of its cardinality.

Allocation. An *allocation* $A \in \{0, 1\}^{n \times m}$ refers to an n-partition $(A_1, \ldots, A_n)$ of $[m]$, where $A_i \subseteq [m]$ is the *bundle* allocated to the agent i. Let $\Pi_n([m])$ denote the set of all n partitions of $[m]$. Given an allocation A, the valuation of an agent $i \in [n]$ for the bundle A_i is $v_i(A_i) = \sum_{j \in A_i} v_{i,j}$. An allocation is said to be *non-wasteful* if no agent is assigned a good that it values at zero, i.e., for each agent i and each good $j \in A_i$, we have $v_{i,j} > 0$.

Nash social welfare. Given an instance $\mathcal{I} = \langle [n], [m], \mathcal{V} \rangle$ and an allocation A, the *Nash social welfare* of A is given by $\text{NSW}(A) \coloneqq \left(\prod_{i \in [n]} v_i(A_i) \right)^{1/n}$. An allocation A^* said to be *Nash optimal* if $A^* \in \arg\max_{A \in \Pi_n([m])} \text{NSW}(A)$. An allocation B is said to be a *β-approximation* (where $0 \leq \beta \leq 1$) for the instance $\mathcal{I}$ if $\text{NSW}(B) \geq \beta \cdot \text{NSW}(A^*)$. For the approximation guarantees to be meaningful, we will assume that the Nash optimal for any given instance has nonzero Nash social welfare.

3 MAIN RESULTS

We provide two main results: a 1.061-approximation algorithm for identical valuations (Theorem 3.1), and an exact algorithm for binary valuations (Theorem 3.2). The proofs of these results are presented in Sections 4 and 5 respectively.

Theorem 3.1 (**Identical valuations**). *Given any fair division instance with additive and identical valuations, there exists a polynomial time* 1.061*-approximation algorithm for the Nash social welfare maximization problem.*

Theorem 3.2 (**Binary valuations**). *Given any fair division instance with additive and binary valuations, a Nash optimal allocation can be computed in polynomial time.*

4 IDENTICAL VALUATIONS: PROOF OF THEOREM 3.1

This section provides the proof of Theorem 3.1.

Theorem 3.1 (**Identical valuations**). *Given any fair division instance with additive and identical valuations, there exists a polynomial time* 1.061*-approximation algorithm for the Nash social welfare maximization problem.*

Our proof of Theorem 3.1 relies on two intermediate results: First, we will show in Lemma 4.1 that the allocation computed by the greedy algorithm (called Alg-Identical, given in Algorithm 1) satisfies an approximate envy-freeness property called EFx, defined below. We will then show in Lemma 4.2 that any allocation with this property—in particular, the allocation computed by Alg-Identical—provides a 1.061 approximation guarantee. We will start by describing the notion of envy-freeness and some of its variants.

Envy-freeness and its variants. Given an instance $\langle [n], [m], \mathcal{V} \rangle$ and an allocation A, we say that an agent $i \in [n]$ *envies* another agent $k \in [n]$ if i prefers the bundle of k over its own bundle, i.e., $v_i(A_k) > v_i(A_i)$. An allocation A is said to be *envy-free* (EF) if each agent prefers its own bundle over that of any other agent, i.e., for every pair of agents $i, k \in [n]$, we have $v_i(A_i) \geq v_i(A_k)$. Likewise, an allocation A is said to be *envy-free up to the least positively*

Algorithm 1: Greedy Algorithm for Identical Valuations (Alg-Identical)

Input: An instance $\langle [n], [m], \mathcal{V} \rangle$ with identical, additive valuations.
Output: An allocation A.

Order the goods in descending order of value, i.e., $v(j_1) \geq v(j_2) \geq \ldots v(j_m) > 0$.
Set $A \leftarrow (\emptyset, \emptyset, \ldots, \emptyset)$.
for $\ell = 1$ *to* m **do**
  Set $i \leftarrow \arg\min_{k \in [n]} v(A_k)$ `// ties are broken lexicographically`
  $A_i \leftarrow A_i \cup \{j_\ell\}$ `// Allocate the good` j_ℓ `to the agent with the least valuation`
end
return A

valued good (EFx) if for every pair of agents $i, k \in [n]$, we have $v_i(A_i) \geq v_i(A_k \setminus \{j\})$ for every $j \in A_k$ such that $v_{i,j} > 0$. The notion of EFx first appeared in the work of Caragiannis et al. [5]. Plaut and Roughgarden [18] study the existence of EFx allocations for special cases of the fair division problem.

We will now describe our algorithm called Alg-Identical.

Greedy algorithm for identical valuations. As mentioned earlier in Section 1, our algorithm Alg-Identical (Algorithm 1) allocates the goods one by one in descending order of their value. In each iteration, a good is assigned to the agent with the least valuation. Assigning goods in this manner ensures that at each step, the algorithm picks the agent providing the greatest improvement in NSW. It is easy to see that Alg-Identical runs in polynomial time. Our next result (Lemma 4.1) shows that Alg-Identical always outputs an EFx allocation.

Lemma 4.1. *The allocation A returned by Alg-Identical is* EFx.

Proof. Let A^ℓ be the allocation maintained by Alg-Identical at the end of the ℓ^{th} iteration. It suffices to show that for each $\ell \in [m]$, if $A^{\ell-1}$ is EFx, then so is A^ℓ.

Recall that Alg-Identical assigns the good j_ℓ to the agent i in the ℓ^{th} iteration, i.e., thus $A_i^\ell = A_i^{\ell-1} \cup \{j_\ell\}$. Thus, only agent i's valuation is affected by the assignment of j_ℓ, while the allocation any other agent $k \in [n] \setminus \{i\}$ is unchanged. Therefore, in order to establish that A^ℓ is EFx, we only need to consider agent i and show that $v(A_i^\ell \setminus \{j\}) \leq v(A_k^\ell)$ for all $k \in [n]$ and each $j \in A_i^\ell$. Since Alg-Identical processes the goods in decreasing order of value, the good j_ℓ is the least valued good in A_i^ℓ. Thus, for any $j \in A_i^\ell$, we have that $v(A_i^\ell \setminus \{j\}) \leq v(A_i^\ell \setminus \{j_\ell\}) = v(A_i^{\ell-1}) \leq v(A_k^\ell)$ for all $k \in [n]$; here, the last inequality follows from the agent selection rule of Alg-Identical, i.e., the fact that $i \in \arg\min_{k \in [n]} v(A_k^{\ell-1})$. This shows that A^ℓ must be EFx. □

Our final result in this section shows that any EFx allocation provides a 1.061 approximation to Nash social welfare when the valuations are additive and identical. Our analysis in Lemma 4.2 is similar to that of Barman et al. [3, Lemma 14], who, under the same set of assumptions, showed that an EF1 allocation[2] provides an $e^{\frac{1}{e}}$-approximation to the Nash social welfare maximization problem.

Lemma 4.2. *Let $\mathcal{I} = \langle [n], [m], \mathcal{V} \rangle$ be an instance with additive and identical valuations, and let A be an* EFx *allocation for $\mathcal{I}$. Then,* $\mathrm{NSW}(A) \geq \frac{1}{1.061}\mathrm{NSW}(A^*)$*, where A^* is the Nash optimal allocation for $\mathcal{I}$.*

Proof. For notational convenience, we will reindex the bundles in the allocation A such that $v(A_1) \geq v(A_2) \geq \cdots \geq v(A_n)$, where v denotes the (additive and identical) valuation function for all agents. Let $\ell := \min_k v(A_k)$ denote the valuation of the least valued bundle under A (thus $v(A_n) = \ell$).

For any agent $k \in [n-1]$ with two or more goods in A_k, EFx property implies that

$$v(A_k) \leq 2\ell. \tag{1}$$

In particular, Equation (1) implies that if $v(A_k) > 2\ell$ for some $k \in [n-1]$, then A_k consists of exactly one good. Let $S := \{k \in [n] : v(A_k) > 2\ell\}$ denote the set of agents with such singleton bundles. Write $s = |S|$ and let $A_S := \{j_1, j_2, \ldots, j_s\}$ denote the set of goods owned by the agents in S.

For analysis, we will now consider a set of allocations where only the goods in A_S are required to be allocated integrally, and any other good can be allocated fractionally among the agents. Formally, we define a *partially-fractional allocation* $B \in [0,1]^{n \times m}$ as follows: For every good $j \in A_S$, $B_{i,j} \in \{0,1\}$ for any agent $i \in [n]$ subject to $\sum_i B_{i,j} = 1$, and for any other good $j \in [m] \setminus A_S$, $B_{i,j} \in [0,1]$ for any agent $i \in [n]$ subject to $\sum_i B_{i,j} = 1$. We let $\mathcal{F}$ denote the set of all such partially-fractional allocations, and let $A^{\mathcal{F}}$ denote the Nash optimal allocation in $\mathcal{F}$.[3] Since all integral allocations belong to $\mathcal{F}$, we have $\mathrm{NSW}(A^{\mathcal{F}}) \geq \mathrm{NSW}(A^*)$. Therefore, in order to prove the lemma, it suffices to show that $\mathrm{NSW}(A) \geq \frac{1}{1.061}\mathrm{NSW}(A^{\mathcal{F}})$.

Define $\alpha := \min_{k \in [n]} v(A_k^{\mathcal{F}})/\ell$. Observe that the goods in A_S, namely $j_1, j_2, \ldots, j_s$, must be allocated to s different agents under $A^{\mathcal{F}}$. This is because the combined value of all goods in $[m] \setminus A_S$ is strictly less than $2\ell(n-s)$. Therefore, if two or more goods in A_S are allocated to the same agent (say agent a), then there must be another agent (agent b) with value strictly less than 2ℓ. In that case, we can obtain another partially-fractional allocation $A' \in \mathcal{F}$ with $\mathrm{NSW}(A') > \mathrm{NSW}(A^{\mathcal{F}})$ by swapping the allocation of agent b under $A^{\mathcal{F}}$ with one of the goods (of value more than 2ℓ) allocated to agent a. This results in a contradiction, since, by assumption, $A^{\mathcal{F}}$ is Nash optimal. Therefore, without loss of generality, no two goods in A_S are allocated to the same agent under $A^{\mathcal{F}}$. This observation allows us to reindex the agents so that $j_i \in A_i^{\mathcal{F}}$ for all $i \in S$.

It is easy to see that $\alpha \geq 1$.[4] In addition, we can show that $\alpha < 2$. Indeed, as argued above, $\sum_{i>s} v(A_i^{\mathcal{F}}) < 2\ell(n-s)$. This implies that $\alpha\ell = \min_i v(A_i^{\mathcal{F}}) < 2\ell$, i.e., $\alpha < 2$. Using this bound we can establish a useful structural property of $A^{\mathcal{F}}$: For all $i \in S$, the bundles

[2] An allocation A is said to be *envy-free up to one good* (EF1) if for every pair of agents $i, k \in [n]$, there exists a good $j \in A_k$ such that $v_i(A_i) \geq v_i(A_k \setminus \{j\})$.

[3] The valuation of an agent under a fractional allocation B is given by $v(B_i) = \sum_j v(j) B_{i,j}$.

[4] If $\alpha < 1$, then it must be that $v(A_n^{\mathcal{F}}) < \ell$, i.e., a nonzero amount of fractional good is taken away from the bundle A_n. In that case, one can reassign (part of) this fractional good—currently assigned to one of the agents in $[n-1]$— to $A_n^{\mathcal{F}}$ and strictly improve the Nash social welfare, contradicting the assumption that $A^{\mathcal{F}}$ is Nash optimal in $\mathcal{F}$.

$A_i^{\mathcal{F}}$ are singletons (i.e., $A_i^{\mathcal{F}} = \{j_i\}$ for all $i \in S$) and, for all $k \notin S$, we have $v(A_k^{\mathcal{F}}) = \alpha\ell$. This follows from the observation that any bundle $A_k^{\mathcal{F}}$ in $A^{\mathcal{F}}$ which has a fractionally allocatable good (say, good j) is of value equal to $\alpha\ell = \min_a v(A_a^{\mathcal{F}})$; otherwise, we can "redistribute" j between $A_k^{\mathcal{F}}$ and $\arg\min_k v(A_k^{\mathcal{F}})$ to obtain another fractional allocation with strictly greater NSW. Moreover, since for any $i \in S$, $j_i \in A_i^{\mathcal{F}}$ and $v(j_i) > \alpha\ell$, the bundle $A_i^{\mathcal{F}}$ does not contain a fractionally allocatable good. This, in particular, implies that $\cup_{i\in S} A_i^{\mathcal{F}} = A_S$. All the remaining goods in $[m] \setminus A_S$ are fractionally allocatable, and hence the bundles $A_k^{\mathcal{F}}$ for all $k \notin S$ are of value equal to $\alpha\ell$. This structural property gives us the following bound for $\text{NSW}(A^{\mathcal{F}})$:

$$\text{NSW}(A^{\mathcal{F}}) = \left(\prod_{i\in S} v(A_i^{\mathcal{F}}) \cdot \prod_{i\in[n]\setminus S} v(A_i^{\mathcal{F}})\right)^{1/n}$$
$$= \left(\prod_{i\in S} v(A_i) \cdot (\alpha\ell)^{(n-s)}\right)^{1/n}. \quad (2)$$

We will now provide a lower bound for $\text{NSW}(A)$ that will allow us to prove the desired approximation guarantee. This is done by constructing an allocation $A' \in \mathcal{F}$ such that $\text{NSW}(A') \leq \text{NSW}(A)$. Along with Equation (2), this provides an analysis-friendly lower bound for the quantity $\text{NSW}(A')/\text{NSW}(A^{\mathcal{F}})$.

We will start with the initialization $A' \leftarrow A$. Next, while there exist two agents $i, k \in [n]$ such that $\ell < v(A'_i) < v(A'_k) < 2\ell$, we transfer goods of value $\Delta = \min\{v(A'_i) - \ell, 2\ell - v(A'_k)\}$ from A'_i (the lesser valued bundle) to A'_k (the larger valued bundle). Such a transfer is possible because the goods in the bundles with value less than 2ℓ are allowed to be allocated fractionally. Notice that the Nash social welfare does not increase as a result of this transfer. Also, it is easy to see that this process terminates, since after each iteration of the while loop, either $v(A'_i) = \ell$ or $v(A'_k) = 2\ell$ or both, and hence some agent can take no further part in any future iterations. Upon termination of the above procedure, there can be at most one agent (say agent r) such that $v(A'_r) \in (\ell, 2\ell)$; for every other agent $k \in [n] \setminus S$, we have $v(A'_k) \in \{\ell, 2\ell\}$.

Let $T = \{k \in [n] : v(A'_k) \geq 2\ell\}$ and let $t = |T|$. Notice that by construction of A', $S \subseteq T$; hence, $s \leq t$. We then have the following bound on the Nash social welfare of the allocation A':

$$\text{NSW}(A') = \left(\prod_{i\in S} v(A'_i) \cdot \prod_{i\in T\setminus S} v(A'_i) \cdot \prod_{i\in[n]\setminus T} v(A'_i)\right)^{1/n}$$
$$\geq \left(\prod_{i\in S} v(A_i) \cdot (2\ell)^{(t-s)} \cdot \ell^{(n-t)}\right)^{1/n}. \quad (3)$$

Let $\phi = \sum_{k\in[n]\setminus S} v(A_k)$ denote the combined value of all goods except for those in A_S. We will now use the allocations $A^{\mathcal{F}}$ and A' to obtain upper and lower bounds for ϕ, which in turn will help us achieve the desired approximation ratio for the allocation A.

First, recall that the goods in the set $A_S = \{j_1, j_2, \ldots, j_s\}$ are allocated as singletons in $A^{\mathcal{F}}$ to the bundles $i \in S$. Along with the fact that $v(A_k^{\mathcal{F}}) = \alpha\ell$ for all $k \in [n] \setminus S$, this gives

$$\phi = \sum_{k\in[n]\setminus S} v(A_k^{\mathcal{F}}) = (n-s)\cdot\alpha\ell. \quad (4)$$

Next, in the allocation A', each bundle corresponding to agents in $T \setminus S$ is valued at exactly 2ℓ, and that for each agent in $[n] \setminus T$ (except for the agent r) is valued at exactly ℓ. By overestimating $v(A'_r)$ to be 2ℓ, we get

$$\phi \leq 2\ell(t+1-s) + \ell(n-t-1). \quad (5)$$

Equations (4) and (5) together imply that

$$\frac{t-s}{n-s} \geq \alpha - 1 - \frac{1}{n-s}. \quad (6)$$

We can lower bound the quantity of interest $\frac{\text{NSW}(A)}{\text{NSW}(A^{\mathcal{F}})}$, as below:

$$\frac{\text{NSW}(A)}{\text{NSW}(A^{\mathcal{F}})} \geq \frac{\text{NSW}(A')}{\text{NSW}(A^{\mathcal{F}})}$$
$$\geq \frac{\left(\prod_{i\in S} v(A_i)\cdot(2\ell)^{(t-s)}\cdot\ell^{(n-t)}\right)^{1/n}}{\left(\prod_{i\in S} v(A_i)\cdot(\alpha\ell)^{(n-s)}\right)^{1/n}}$$

(from Equations (2) and (3))

$$= \left(\frac{2^{t-s}}{\alpha^{n-s}}\right)^{1/n}$$
$$\geq \left(\frac{2^{t-s}}{\alpha^{n-s}}\right)^{1/(n-s)}$$
$$\left(\text{since } \left(\frac{2^{t-s}}{\alpha^{n-s}}\right)^{1/n} \leq \frac{\text{NSW}(A)}{\text{NSW}(A^{\mathcal{F}})} \leq 1 \implies \frac{2^{t-s}}{\alpha^{n-s}} \leq 1\right)$$
$$\geq \frac{2^{\alpha-1-\frac{1}{n-s}}}{\alpha} \quad \text{(from Equation (6))}. \quad (7)$$

The $\frac{1}{n-s}$ term in the exponent in Equation (7) can be neglected via a scaling argument as follows: Imagine constructing a *scaled-up* instance $\mathcal{I}'$ consisting of $c \geq 1$ copies of the instance $\mathcal{I}$. Each agent's valuation for (a copy of) a good in the instance $\mathcal{I}'$ is exactly as in the original instance $\mathcal{I}$; thus, $\mathcal{I}'$ has identical valuations. For any allocation A that is EFx for $\mathcal{I}$, the allocation $B = (A, A, \ldots, A)$ is EFx for $\mathcal{I}'$. Let $n', s', \alpha', \ell', A'^{\mathcal{F}}$ denote the analogues of $n, s, \alpha, \ell, A^{\mathcal{F}}$ in $\mathcal{I}'$. Also, let $\tilde{A}$ denote the fractional allocation $(A^{\mathcal{F}}, A^{\mathcal{F}}, \ldots, A^{\mathcal{F}})$ in $\mathcal{I}'$. It is easy to see that $n' = cn$, $s' = cs$, $\alpha' = \alpha$, and $\ell' = \ell$. Moreover,

$$\frac{\text{NSW}(A)}{\text{NSW}(A^{\mathcal{F}})} = \frac{\text{NSW}(B)}{\text{NSW}(\tilde{A})} \geq \frac{\text{NSW}(B)}{\text{NSW}(A'^{\mathcal{F}})} \geq \frac{2^{\alpha-1-\frac{1}{c(n-s)}}}{\alpha},$$

where the first term is for the instance $\mathcal{I}$, and the remaining terms are for the instance $\mathcal{I}'$. In addition, the relation $\text{NSW}(A'^{\mathcal{F}}) \geq \text{NSW}(\tilde{A})$ follows from the optimality of $A'^{\mathcal{F}}$ for $\mathcal{I}'$. Notice that the agent with the least valuation in $\mathcal{I}$ (under the allocation A) values its bundle at strictly below 2ℓ, and thus $s < n$. Therefore, the quantity $n' - s' = c(n-s)$ can be made arbitrarily large for an appropriate choice of c, allowing us to ignore the $\frac{1}{n-s}$ term in Equation (7).

We therefore have that $\frac{\text{NSW}(A)}{\text{NSW}(A^{\mathcal{F}})} \geq \frac{2^{\alpha-1}}{\alpha}$. The function $\frac{2^{\alpha-1}}{\alpha}$ for $\alpha \geq 0$ is minimized at $\alpha = 1/\ln 2 \approx 1.44$, and the minimum value is $\frac{1}{2}e\ln 2 \approx \frac{1}{1.061}$. This gives the desired approximation ratio, completing the proof of Lemma 4.2. □

Example 4.3 shows that the approximation guarantee of Lemma 4.2 is almost tight.

Example 4.3. Consider an instance with m goods (m is even) and $n = 2$ agents, where the (additive and identical) valuations are given by $v(j_1) = v(j_2) = m-2$, and $v(j_\ell) = 1$ for $\ell \in \{3, 4, \ldots, m\}$. Notice that the allocation $A = \{(j_1, j_2), (j_3, \ldots, j_m)\}$ is EFx. Additionally, $\text{NSW}(A) = ((2m-4)\cdot(m-2))^{1/2}$. It is also clear that $\text{NSW}(A^*) = \frac{3}{2}(m-2)$. The approximation ratio of A is given by

$$\frac{\text{NSW}(A)}{\text{NSW}(A^*)} = \left(\frac{(2(m-2))\cdot(m-2)}{(3(m-2)/2)\cdot(3(m-2)/2)}\right)^{1/2} \approx \frac{1}{1.0607},$$

which closely matches the approximation guarantee of Lemma 4.2.

5 BINARY VALUATIONS: PROOF OF THEOREM 3.2

This section provides the proof of Theorem 3.2.

Theorem 3.2 (**Binary valuations**). *Given any fair division instance with additive and binary valuations, a Nash optimal allocation can be computed in polynomial time.*

Our proof of Theorem 3.2 relies on a greedy algorithm (Algorithm 2, hereafter referred to as Alg-Binary). Starting from any suboptimal allocation, Alg-Binary identifies a pair of agents such that a chain of swaps between them provides the greatest improvement in Nash social welfare (from among all pairs of agents). Lemma 5.1 quantifies the progress toward the Nash optimal allocation made by Alg-Binary in each step. As it turns out, the algorithm is required to run for at most $2m(n+1)\ln(nm)$ iterations. Overall, this provides a polynomial time algorithm for computing a Nash optimal allocation for binary valuations.

Greedy algorithm for binary valuations. The input to Alg-Binary is an instance with additive and binary valuations along with a suboptimal allocation, and output is a Nash optimal allocation. At each step, the algorithm performs a greedy local update over the current allocation. Specifically, given an allocation $A = (A_1, A_2, \ldots, A_n)$, Alg-Binary constructs a directed graph $G(A)$ as follows: There is a vertex for each agent (hence n vertices overall), and between any pair of vertices u and v, there are $|\Gamma_v \cap A_u|$ parallel edges directed from u to v.[5] A directed edge (u, v) exists if and only if there exists a good that is valued by v and is currently assigned to u. Observe that a directed simple path $P = (u_1, u_2, \ldots, u_k)$ in $G(A)$ corresponds to a sequence of reallocations. For each directed edge (u_i, u_{i+1}), there exists a good $j \in A_{u_i}$ that can be reassigned to u_{i+1} via the updates $A_{u_i} \leftarrow A_{u_i} \setminus \{j\}$ and $A_{u_{i+1}} \leftarrow A_{u_{i+1}} \cup \{j\}$.

Let $A(P)$ denote the allocation obtained by reallocating goods along the path $P = (u_1, u_2, \ldots, u_k)$. Such a reallocation increases (respectively, decreases) the valuation of u_k (respectively, u_1) by one, while the valuations of all intermediate agents $u_2, \ldots, u_{k-1}$ are unchanged. The algorithm Alg-Binary greedily selects a specific path P in $G(A)$, and reallocates the goods along P to obtain the allocation $A' := A(P)$. Lemma 5.1 below describes the progress toward the optimal solution made by such a reallocation.

[5] Recall that $\Gamma_i := \{j \in [m] : v_{i,j} > 0\}$.

Algorithm 2: Greedy Algorithm for Binary Valuations (Alg-Binary)

Input: An instance $\langle [n], [m], \mathcal{V}\rangle$ with binary, additive valuations, and an allocation A.
Output: An allocation A'.

Set $A^0 \leftarrow A$.
for $i = 1$ *to* $2m(n+1)\ln(nm)$ **do**
  Construct the graph $G(A^{i-1})$ for the current allocation A^{i-1}.
  $R \leftarrow \{(u, v) : v \text{ is reachable from } u \text{ in } G(A^{i-1})\}$.
  for each $(u, v) \in R$ **do**
    $A^{i-1}(u, v) \leftarrow$ The allocation obtained by reallocating along some path from u to v.
  endfor
  if $\max_{(u,v)\in R} \text{NSW}(A^{i-1}(u,v)) > \text{NSW}(A^{i-1})$ **then**
    Update $A^i \leftarrow \underset{A^{i-1}(u,v)\,:\,(u,v)\in R}{\arg\max} \text{NSW}(A^{i-1}(u,v))$.
  else
    return A^{i-1}
  end
end

Lemma 5.1. *Given a suboptimal allocation A, there exist agents u and v such that v is reachable from u in G(A), and reallocating along* ***any*** *directed path P from u to v leads to an allocation $A' := A(P)$ that satisfies (here A^* denotes the Nash optimal allocation)*

$$\ln\text{NSW}(A^*) - \ln\text{NSW}(A') \leq \left(1 - \frac{1}{m}\right)\left(\ln\text{NSW}(A^*) - \ln\text{NSW}(A)\right).$$

Remark 1. Note that there can be multiple paths P from u to v in $G(A)$, and more than one good can be reallocated along a fixed edge of P, which might lead to different allocations $A(P)$. However, the Nash social welfare of *any* resulting allocation is the same, since the valuation of u (respectively, v) goes down (respectively, up) by one and that of every other agent remains the same. Hence, the choice of path between a fixed pair of vertices is inconsequential.

We will now show that Lemma 5.1 can used to prove Theorem 3.2, followed by a proof of Lemma 5.1.

Proof of Theorem 3.2. Lemma 5.1 ensures that if there does not exist an improving reallocation, then the current allocation A^{i-1} is optimal. Hence, for the rest of the proof, we will focus on the case wherein the for-loop executes for all $2m(n+1)\ln(nm)$ steps.

The update rule followed by Alg-Binary and Lemma 5.1 together guarantee that at the end of iteration i, we have

$$\ln\text{NSW}(A^*) - \ln\text{NSW}(A^i) \leq \left(1 - \frac{1}{m}\right)(\ln\text{NSW}(A^*) - \ln\text{NSW}(A^{i-1})).$$

Repeated use of the above bound gives

$$\ln\text{NSW}(A^*) - \ln\text{NSW}(A^i) \leq$$

$$\left(1-\frac{1}{m}\right)^{i}(\ln \mathrm{NSW}(A^*)-\ln \mathrm{NSW}(A^0)).$$

Since Alg-Binary executes for $2m(n+1)\ln(mn)$ iterations, the difference between the optimal allocation A^* and the allocation A' returned by the algorithm is given by

$$\begin{aligned}
&\ln \mathrm{NSW}(A^*)-\ln \mathrm{NSW}(A')\\
&\leq \left(1-\frac{1}{m}\right)^{2m(n+1)\ln(nm)}(\ln \mathrm{NSW}(A^*)-\ln \mathrm{NSW}(A^0))\\
&\leq \frac{1}{e^{2(n+1)\ln(nm)}}(\ln \mathrm{NSW}(A^*)-\ln \mathrm{NSW}(A^0))\\
&\leq \frac{1}{(nm)^{2(n+1)}}\ln \mathrm{NSW}(A^*)\\
&\leq \frac{\ln m}{(nm)^{2(n+1)}}\\
&\qquad \text{(since } \mathrm{NSW}(A^*)\leq m \text{ for binary valuations)}\\
&\leq \frac{1}{nm^{2n}} \quad \text{(since } \ln m\leq m\text{, and } n,m\geq 2)\\
&< \frac{1}{n}\ln\left(1+\frac{1}{m^n}\right)\\
&\qquad \text{(since } \ln(1+x)>x^2 \text{ for } x\in(0,0.5)).
\end{aligned}$$

Thus, $\prod_{i\in[n]} v_i(A_i^*) < \prod_{i\in[n]} v_i(A_i')\left(1+\frac{1}{m^n}\right)$. We already know that $\prod_{i\in[n]} v_i(A_i') \leq \prod_{i\in[n]} v_i(A_i^*)$. Since the valuations are assumed to be integral, and $\prod_{i\in[n]} v_i(A_i') \leq m^n$, we have that $\mathrm{NSW}(A^*) = \mathrm{NSW}(A')$. Hence, A' is Nash optimal. □

We will now provide a proof of Lemma 5.1.

Proof of Lemma 5.1. Our proof of existence of the desired path P in the graph $G(A)$ is made convenient by the formulation of another graph $G^*(A)$. This graph is utilized only in the analysis of the algorithm and never explicitly constructed.

Recall that A^* refers to a Nash optimal allocation. Consider the directed graph $G^*(A)$ consisting of n vertices, one for each agent, and a directed edge (u,v) for each good $j\in A_u\cap A_v^*$. The edge (u,v) indicates that the good j must be transferred from u to v to reach the optimal allocation A^*. Note that the total number of edges in $G^*(A)$ is at most m.

Besides defining the graph $G^*(A)$, we will also classify the agents depending on their valuation relative to A^*. In particular, let $\mathcal{E}$ and $\mathcal{D}$ denote the set of agents with *excess* and *deficit* valuations respectively, i.e., $\mathcal{E} := \{u\in[n] : |A_u| > |A_u^*|\}$ and $\mathcal{D} := \{v\in[n] : |A_v| < |A_v^*|\}$.[6] Any agent $t\in[n]\setminus(\mathcal{E}\cup\mathcal{D})$ satisfies $|A_t| = |A_t^*|$.

The remainder of the proof consists of two parts: First, we will show that the edge set of $G^*(A)$ can be partitioned into simple directed paths $\mathcal{P}=\{P_1,P_2,\ldots,P_k\}$ and cycles $C=\{C_1,C_2,\ldots\}$ such that each path $P_i\in\mathcal{P}$ starts at a vertex in $\mathcal{E}$ and ends at a vertex in $\mathcal{D}$. Second, we will use this decomposition to argue that one of the paths $P_i\in\mathcal{P}$ leads to an allocation $A' := A(P_i)$ that satisfies the bound in Lemma 5.1. The lemma will then follow by observing that the edges of P_i are also contained in the graph $G(A)$ constructed by Alg-Binary. Note that the existence of P_i shows that the end vertex of P_i (say, v) is reachable from the start vertex of P_i (say, u) in $G(A)$. As noted earlier in Remark 1, reallocating along *any* path between u and v leads to the stated improvement in Nash social welfare.

We will start by proving the claim about decomposition of the edge set of $G^*(A)$. Consider a graph H^* where for each vertex u of $G^*(A)$, we include max(indegree(u), outdegree(u)) vertices, say $\{u^1,u^2,\ldots\}$. Suppose the vertex u has ℓ incoming edges and ℓ' outgoing edges in $G^*(A)$. To construct H^*, first we pick an arbitrary one-to-one assignment between the incoming edges and $\{u^1,u^2,\ldots,u^\ell\}$. Similarly, each outgoing edge gets uniquely assigned to one of the vertices in $\{u^1,u^2,\ldots,u^{\ell'}\}$. With these assignments in hand, for every directed edge $e=(u,v)$ in $G^*(A)$, we include a directed edge (u^i,v^j) in H^* if and only if e is assigned to u^i and v^j. It is easy to see that each edge in H^* corresponds to an edge in $G^*(A)$ and vice versa.

Notice that each vertex in H^* has at most one incoming and at most one outgoing edge. Furthermore, if u^i is a source in H^*, then $u\in\mathcal{E}$. Similarly, if v^j is a sink in H^*, then $v\in\mathcal{D}$. These properties together imply that the edges in H^* can be partitioned into paths and cycles such that each path starts at a vertex u^i with $u\in\mathcal{E}$ and ends at a vertex v^j with $v\in\mathcal{D}$. The correspondence between the edges of H^* and $G^*(A)$ gives us the desired collection of paths $\mathcal{P}=\{P_1,P_2,\ldots,P_k\}$ and cycles C in $G^*(A)$.[7] The aforementioned properties also imply that the paths in H^* are edge-disjoint, therefore $k\leq m$.

We will now show that for one of the paths $P_i\in\mathcal{P}$ in $G^*(A)$ (and therefore, also in $G(A)$), the allocation $A' := A(P_i)$ achieves the bound in Lemma 5.1. First, observe that reallocating along a cycle in $G^*(A)$ does not change the Nash social welfare. Hence, in order to reach a Nash optimal allocation starting from A, it suffices to reallocate goods along the paths $P_1,P_2,\ldots,P_k$. Moreover, since the paths in $\mathcal{P}$ are edge disjoint in the graph H^*, they correspond to reallocation of disjoint sets of goods. This means that the reallocations corresponding to a path $P_i\in\mathcal{P}$ can be performed independently of those corresponding to another path $P_j\in\mathcal{P}$.

Next, consider the sequence of allocations $B^1,B^2,\ldots,B^k$, obtained by successively reallocating along the paths $P_1,P_2,\ldots,P_k$. That is, $B^1=A(P_1)$, $B^2=B^1(P_2)$, and so on. Thus, the allocation B^k must be Nash optimal, i.e., $\mathrm{NSW}(A^*)=\mathrm{NSW}(B^k)$. Consider the telescoping sum given by $\ln \mathrm{NSW}(A^*)-\ln \mathrm{NSW}(A) = \sum_{i=1}^{k-1}\ln \mathrm{NSW}(B^i)-\ln \mathrm{NSW}(B^{i-1})$, where $B^0=A$. Since $k\leq m$, there must exist $i\in[k]$ such that

$$\ln \mathrm{NSW}(B^i)-\ln \mathrm{NSW}(B^{i-1}) \geq \frac{1}{m}\left(\ln \mathrm{NSW}(A^*)-\ln \mathrm{NSW}(A)\right). \quad (8)$$

We will now show that the allocation $A' := A(P_i)$ satisfies

$$\ln \mathrm{NSW}(A')-\ln \mathrm{NSW}(A) \geq \ln \mathrm{NSW}(B^i)-\ln \mathrm{NSW}(B^{i-1}). \quad (9)$$

Indeed, recall that each path in $\mathcal{P}$ starts at a vertex in $\mathcal{E}$ and ends at a vertex in $\mathcal{D}$. Hence, as we proceed through reallocations corresponding to $P_1,\ldots,P_k$, the cardinality of the set of goods assigned to any agent $u'\in\mathcal{E}$ is non-increasing and that of $v'\in\mathcal{D}$ is non-decreasing. Therefore, if u (respectively, v) is the start (respectively, end) vertex of P_i, then $k_u\geq k'_u$ and $k_v\leq k'_v$, where k_u, k_v, k'_u

[6]For binary valuations and a non-wasteful allocation A, we have $v_i(A)=|A|$ for each agent $i\in[n]$.

[7]We can ensure that the paths in $\mathcal{P}$ are *simple* by removing cycles from each P_i and placing such cycles in C.

and k'_v are the number of goods assigned to u and v in allocations A and B^{i-1} respectively. Since $\ln \text{NSW}(B^i) - \ln \text{NSW}(B^{i-1}) = \ln(k'_u - 1) + \ln(k'_v + 1) - (\ln k'_u + \ln k'_v)$ and $\ln \text{NSW}(A') - \ln \text{NSW}(A) = \ln(k_u - 1) + \ln(k_v + 1) - (\ln k_u + \ln k_v)$, the concavity of $\ln(\cdot)$ implies Equation (9). Finally, Equations (8) and (9) give

$$\ln \text{NSW}(A^*) - \ln \text{NSW}(A') \leq \left(1 - \frac{1}{m}\right)(\ln \text{NSW}(A^*) - \ln \text{NSW}(A)),$$

as desired. □

Notice that the proof of Lemma 5.1 works exactly the same way when for each agent i, $v_i(A_i) = f_i(|A_i|)$ for some *concave* function f_i. That is, the valuation of an agent can be an (agent-specific) concave function of the cardinality (i.e., the number of nonzero valued goods owned by the agent). Thus, Alg-Binary can find a Nash optimal allocation in polynomial time even when the valuation functions of agents are concave in cardinality. This observation is formalized in Corollary 5.2.

Corollary 5.2. *Given any fair division instance with concave and binary valuations, a Nash optimal allocation can be computed in polynomial time.*

Remark 2. A well-studied class of valuation functions captured by Corollary 5.2 is that of *budget-additive* valuations [13]. Under this class, the valuation of an agent $i \in [n]$ for a set of goods $G \subseteq [m]$ is given by $v_i(G) := \min\{c_i, \sum_{j \in G} v_{i,j}\}$, where $c_i > 0$ is an (agent-specific) constant, known as the *utility cap*.

Garg et al. [10] recently gave a $(2.404 + \varepsilon)$-approximation algorithm for maximizing Nash social welfare under budget-additive valuations (for any $\varepsilon > 0$). For binary valuations, a budget-additive valuation function turns out to be a special case of the concave-in-cardinality functions mentioned above. Hence, by Corollary 5.2, a Nash optimal allocation can be found in polynomial time when the valuations are binary and budget-additive. It is unclear whether the existing techniques for finding a Nash optimal allocation under binary and additive valuations [8] admit a similar generalization.

ACKNOWLEDGMENTS

We thank the anonymous reviewers for their careful reading of the paper and several helpful comments. SB was supported in part by a Ramanujan Fellowship (SERB - SB/S2/RJN-128/2015). RV thanks travel grants from Google and Microsoft Research to present this work at AAMAS 2018.

REFERENCES

[1] Nima Anari, Shayan Oveis Gharan, Tung Mai, and Vijay V Vazirani. 2018. Nash Social Welfare for Indivisible Items under Separable, Piecewise-Linear Concave Utilities. In *Proceedings of the 29th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*. 2274–2290.

[2] Nima Anari, Shayan Oveis Gharan, Amin Saberi, and Mohit Singh. 2017. Nash Social Welfare, Matrix Permanent, and Stable Polynomials. In *Proceedings of the 8th Conference on Innovations in Theoretical Computer Science (ITCS)*.

[3] Siddharth Barman, Sanath Krishnamurthy, and Rohit Vaish. 2017. Finding Fair and Efficient Allocations. *arXiv preprint arXiv:1707.04731* (2017).

[4] Xiaohui Bei, Jugal Garg, Martin Hoefer, and Kurt Mehlhorn. 2017. Earning Limits in Fisher Markets with Spending-Constraint Utilities. In *Proceedings of the International Symposium on Algorithmic Game Theory (SAGT)*. 67–79.

[5] Ioannis Caragiannis, David Kurokawa, Hervé Moulin, Ariel D Procaccia, Nisarg Shah, and Junxing Wang. 2016. The Unreasonable Fairness of Maximum Nash Welfare. In *Proceedings of the 2016 ACM Conference on Economics and Computation (EC)*. 305–322.

[6] Richard Cole, Nikhil Devanur, Vasilis Gkatzelis, Kamal Jain, Tung Mai, Vijay V Vazirani, and Sadra Yazdanbod. 2017. Convex Program Duality, Fisher Markets, and Nash Social Welfare. In *Proceedings of the 2017 ACM Conference on Economics and Computation (EC)*. 459–460.

[7] Richard Cole and Vasilis Gkatzelis. 2015. Approximating the Nash Social Welfare with Indivisible Items. In *Proceedings of the Forty-Seventh Annual ACM on Symposium on Theory of Computing (STOC)*. 371–380.

[8] Andreas Darmann and Joachim Schauer. 2015. Maximizing Nash Product Social Welfare in Allocating Indivisible Goods. *European Journal of Operational Research* 247, 2 (2015), 548–559.

[9] Edmund Eisenberg and David Gale. 1959. Consensus of Subjective Probabilities: The Pari-mutuel Method. *The Annals of Mathematical Statistics* 30, 1 (1959), 165–168.

[10] Jugal Garg, Martin Hoefer, and Kurt Mehlhorn. 2018. Approximating the Nash Social Welfare with Budget-Additive Valuations. In *Proceedings of the 29th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*. 2326–2340.

[11] Mamoru Kaneko and Kenjiro Nakamura. 1979. The Nash Social Welfare Function. *Econometrica: Journal of the Econometric Society* (1979), 423–435.

[12] Euiwoong Lee. 2017. APX-hardness of Maximizing Nash Social Welfare with Indivisible Items. *Inform. Process. Lett.* 122 (2017), 17–20.

[13] Benny Lehmann, Daniel Lehmann, and Noam Nisan. 2001. Combinatorial Auctions with Decreasing Marginal Utilities. In *Proceedings of the 3rd ACM Conference on Electronic Commerce (EC)*. 18–28.

[14] Hendrik W Lenstra Jr. 1983. Integer Programming with a Fixed Number of Variables. *Mathematics of Operations Research* 8, 4 (1983), 538–548.

[15] Hervé Moulin. 2004. *Fair Division and Collective Welfare*. MIT press.

[16] John F Nash Jr. 1950. The Bargaining Problem. *Econometrica: Journal of the Econometric Society* (1950), 155–162.

[17] Trung Thanh Nguyen and Jörg Rothe. 2014. Minimizing Envy and Maximizing Average Nash Social Welfare in the Allocation of Indivisible Goods. *Discrete Applied Mathematics* 179 (2014), 54–68.

[18] Benjamin Plaut and Tim Roughgarden. 2018. Almost Envy-freeness with General Valuations. In *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*. SIAM, 2584–2603.

[19] Magnus Roos and Jörg Rothe. 2010. Complexity of Social Welfare Optimization in Multiagent Resource Allocation. In *Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*. 641–648.

[20] Hal R Varian. 1974. Equity, Envy, and Efficiency. *Journal of Economic Theory* 9, 1 (1974), 63–91.