Vrije Universiteit Brussel

Fair allocation of indivisible goods with minimum inequality or minimum envy

Cornilly, Dries; Puccetti, Giovanni; Rüschendorf, Ludger; Vanduffel, Steven

Published in:
European Journal of Operational Research

DOI:
10.1016/j.ejor.2021.06.020

Publication date:
2022

License:
CC BY-NC-ND

Document Version:
Accepted author manuscript

Link to publication

Citation for published version (APA):
Cornilly, D., Puccetti, G., Rüschendorf, L., & Vanduffel, S. (2022). Fair allocation of indivisible goods with minimum inequality or minimum envy. *European Journal of Operational Research*, *297*(2), 741-752. https://doi.org/10.1016/j.ejor.2021.06.020

Copyright
No part of this publication may be reproduced or transmitted in any form, without the prior written permission of the author(s) or other rights holders to whom publication rights have been transferred, unless permitted by a license attached to the publication (a Creative Commons license or other), or unless exceptions to copyright law apply.

Take down policy
If you believe that this document infringes your copyright or other rights, please contact openaccess@vub.be, with details of the nature of the infringement. We will investigate the claim and if justified, we will take the appropriate steps.

Download date: 25. Apr. 2024

Fair allocation of indivisible goods with minimum inequality or minimum envy

Dries Cornilly,* Giovanni Puccetti,† Ludger Rüschendorf,‡ and Steven Vanduffel.§

June 9, 2021

Abstract

We provide a new mathematical framework for the classic problem of fair allocation of indivisible goods, showing that it can be formulated as the problem of finding an optimal column rearrangement of multiple matrices. Based on this formulation, we design two novel algorithms called MinCov and MinCovTarget to find optimal allocations under the newly introduced notion of minimum social inequality, and the popular notion of minimum envy. Numerical illustrations show an excellent performance of the newly developed algorithms also with respect to other allocation criteria, in particular for the maximum Nash welfare.

Key-words: decision analysis, algorithms for fair allocation, perfect social equality, minimum envy, maximum Nash welfare, Pareto optimality.

*Dries Cornilly (email: `dcornilly@asteria-im.com`). Asteria Investment Managers SA, Rue du Rhône 62, 1204 Genève, Switzerland.

†Giovanni Puccetti (email: `giovanni.puccetti@unimi.it`). University of Milan, Via Conservatorio 7, 20122 Milano, Italy.

‡Ludger Rueschendorf (e-mail: `ruschen@stochastik.uni-freiburg.de`). University of Freiburg, Ernst-Zermelo-Straße 1, 79104 Freiburg, Germany

§Steven Vanduffel, corresponding author (e-mail: `steven.vanduffel@vub.ac.be`). Vrije Universiteit Brussel, Pleinlaan 2, 1050, Brussels, Belgium.

Electronic copy available at: https://ssrn.com/abstract=3512113

1 Introduction

Fair allocation (also referred to as fair division) is a classic and relevant problem at the interface of mathematics, economics, and game theory. It has a long history and extensive literature, for which we refer for instance to Chevaleyre et al. (2017), Kurokawa et al. (2018), and Moulin (2019). Fair allocation problems are generally distinguished into the two cases of divisible and indivisible goods. The literature on cake cutting, for instance, is concerned with divisible goods (Brams and Taylor 1996, Robertson and Webb 1998), but in this paper we focus on the more challenging case when the items to be allocated are indivisible (Bouveret et al. 2016, Lang and Rothe 2016).

By *fair allocation of indivisible goods* we refer in the following to the problem of allocating a finite set of desirable goods to a finite number of agents. Each object is assumed to be *indivisible* (it loses its value if broken or divided into pieces) and *non-sharable* (the same item cannot be allocated to more than one agent). The need for fair allocation of a set of objects among several agents arises naturally in many scenarios, from inheritance and divorce settlements to border disputes and medical supplies.

The purpose of fair allocation is to find an allocation among the agents that satisfies an a-priori fixed optimality property, which is referred to as a *fairness criterion*. In the specialized literature, there exists a broad variety of fairness notions; see for instance Thomson (2011), Lang and Rothe (2016), and Bouveret and Lemaître (2016) for surveys. In this paper, we focus on the widely used notion of *envy* and on the newly introduced criterion of *minimum social inequality*.

An allocation is said to be envy-free (Foley 1967) if no agent prefers the bundle of goods received by any other agent to the own one. The notion of envy-freeness is compelling, but an envy-free allocation might not exist in practice (just consider the problem of allocating a diamond and a rock to two agents). In the cases where a null level of envy is not attainable, it still makes sense to minimize the maximal envy among any pair of agents (Lipton et al. 2004). A minimum envy allocation always exists (but might be difficult to find).

In this paper, we introduce a novel notion of fairness which consists in minimizing the social inequality of the economic system, that is in making the utilities gained by the agents as similar as possible (under each agent's belief). When each agent believes that all agents have gained exactly the same level of utility, we speak of *perfect social equality*. Similarly to a no-envy allocation, a perfect social equilibrium is not always attainable. In the latter case, one can aim to obtain the allocation with the minimum possible level of inequality, i.e., the allocation yielding the minimum possible variance of the utilities allocated to each agent.

We show that the two notions of envy and social equality have interesting mathematical relationships. It is immediate to see that a perfect social equality allocation is also envy-free. In general, we establish that the level of envy present in a minimum inequality allocation is bounded from above by the level of inequality.

When there are exactly two players, a variety of practical methods for allocating indivisible

Electronic copy available at: https://ssrn.com/abstract=3512113

goods are available; see the comparison study given in Kilgour and Vetschera (2018). General fair allocation algorithms for fairness criteria different from those treated here are also described in Vetschera (2010), Caragiannis et al. (2019), Kurokawa et al. (2018), and Ghodsi et al. (2018). Lipton et al. (2004) provide a polynomial time algorithm that computes allocations with bounded maximum envy and an algorithm for the so-called minimum envy-ratio problem. These authors notice that finding an allocation with minimum possible envy is in general not solvable in polynomial time unless $\mathcal{P} = \mathcal{NP}$; see also Bouveret et al. (2005).

In this paper, we deviate from the existing literature by reformulating the problem of fair allocation as a rearrangement problem of multiple matrices. The rationale behind our new mathematical framework is to replace the notion of an allocation of goods with that of a column rearrangement of a finite set of matrices. This reformulation leads us to provide two novel numerical algorithms, called MinCov and MinCovTarget. The MinCov algorithm is capable of finding an allocation with minimum (null) social inequality whereas the MinCovTarget introduces a target parameter which allows us to interpolate between the minimum social inequality and the minimum envy criterion.

The numerical results shown in this paper are promising and are also supplemented with some theoretical evidence. We show excellent figures for the criteria of minimum social inequality (MinCov) and minimal envy (MinCovTarget). With respect to minimal envy, allocations provided by the MinCovTarget algorithm are at least as good as those delivered by approximate maximum Nash welfare (MNW) allocations, and in some cases outperforming MNW. Allocations provided by MinCovTarget also indicate approximate Pareto-optimality for moderate target values.

If compared to existing approaches, our algorithms are straightforward to implement, fast, and do not have restrictions on the number of agents/goods considered. The interest reader can retrieve the code for the applications shown in this paper at

`https://github.com/cdries/FAIG`.

2 Fair allocation as a rearrangement problem

Following Chevaleyre et al. (2017) and Bouveret and Lemaître (2016), from which we slightly adapt the notation, we define in what follows an additive MultiAgent Resource Allocation (a-MARA) as a triple setting $(\mathcal{N}, \mathcal{D}, \mathbf{X})$ where $\mathcal{N} = \{1, \ldots, n\}$ is a finite set of $n \geqslant 2$ agents, $\mathcal{D} = \{1, \ldots, d\}$ is a finite set of $d \geqslant 1$ indivisible and non-shareable objects, and $\mathbf{X}$ is a matrix representing the additive utility function of each agents over the set of goods. These assumptions apply to a broad range of real-world situations such as some inheritance disputes where the items are indivisible, of different values to each of the parties, and the agents might not be able to compensate the others financially. Notice that these assumptions do not prevent the treatment for instance of medical divisions receiving different units of consumptions goods as these kinds of problems can always be modelled with non-sharable goods by introducing several units of the same good.

Electronic copy available at: https://ssrn.com/abstract=3512113

Denote by $\mathcal{P}(\{1,\dots,d\})$ the set of all subsets of $\{1,\dots,d\}$. In what follows, any such subset of $\{1,\dots,d\}$ is called a *bundle*. An *allocation* $\kappa : \{1,\dots,n\} \to \mathcal{P}(\{1,\dots,d\})$ maps agent i to the bundle $\kappa(i)$ he receives, satisfying $\kappa(i) \cap \kappa(j) = \emptyset, i \neq j$, and $\cup_{i=1}^{n} \kappa(i) = \{1,\dots,d\}$ (each good is owned by exactly one agent, and nothing is thrown away). Agents' valuations of the goods are stored in a $n \times d$ value matrix $\mathbf{X} = (x_{ij})$, in which $x_{ij} > 0$ is the utility given by agent $i \in \{1,\dots,n\}$ to object $j \in \{1,\dots,d\}$. We use positive values $x_{ij} > 0$ without loss of generality, since one can always add a constant C (e.g. $C = -\min x_{ij} + 1$) to all entries of $X = (x_{ij})$. Specifically, we assume that agents' preferences are cardinal and additive, meaning that the utility of agent i in receiving the bundle $\kappa(i)$ is given by

$$u_i(\kappa(i)) = \sum_{j \in \kappa(i)} x_{ij}.$$

Now, consider the trivial allocation in which the first agent receives all goods (hence all other agents receive nothing) and define the n matrices $\mathbf{X}^k = (x_{ij}^k), k = 1,\dots,n$, as

$$x_{ij}^k = \begin{cases} x_{kj}, & \text{if } i = 1, \\ 0, & \text{if } i \neq 1. \end{cases} \tag{1}$$

Notice that the k-th matrix $\mathbf{X}^k$ contains only valuations made by the k-th agent. In particular, the first row of $\mathbf{X}^k$ contains the values that the $k-$th agent assigns to the goods received by the first agent, while all other rows contains only nulls (under this allocation, the first agent receives all goods and all other agents receive nothing).

Example 2.1. We provide an illustration of our set-up for $d = 4$ objects to be allocated to $n = 2$ agents. The first row of the value matrix $\mathbf{X}$ contains the valuations of the items made by the first agent and the second row contains those of the second agent. Assume $\mathbf{X}$ is given by

$$\mathbf{X} = \begin{pmatrix} 6 & 10 & 9 & 8 \\ 5 & 12 & 13 & 4 \end{pmatrix}.$$

The matrices $\mathbf{X}^1$ and $\mathbf{X}^2$ correspond to the trivial allocation $\kappa(1) = \{1,2,3,4\}$ (the first agent receives all goods) and $\kappa(2) = \emptyset$ (the second one receives nothing). They are thus given by

$$\mathbf{X}^1 = \begin{pmatrix} 6 & 10 & 9 & 8 \\ 0 & 0 & 0 & 0 \end{pmatrix}, \quad \mathbf{X}^2 = \begin{pmatrix} 5 & 12 & 13 & 4 \\ 0 & 0 & 0 & 0 \end{pmatrix}.$$

Now, let $\mathcal{A}$ be the set of all vectors $\pi = (\pi_1,\dots,\pi_d)$ in which the components π_j are permutations

Electronic copy available at: https://ssrn.com/abstract=3512113

of $\{1, \ldots, n\}$. Given a vector of permutations $\pi \in \mathcal{A}$, define $\mathbf{X}^{k,\pi} = (x_{ij}^{k,\pi})$ by

$$x_{ij}^{k,\pi} = x_{\pi_j(i)j}^{k}, \tag{2}$$

i.e., each π_j permutes the elements of the j-th column of all matrices $\mathbf{X}^k$ to obtain the matrices $\mathbf{X}^{k,\pi}, k = 1, \ldots, n$. In what follows, we call any $\pi \in \mathcal{A}$ a *rearrangement* and each matrix $\mathbf{X}^{k,\pi}$ a *rearranged matrix*.

Denote by $s_i^{k,\pi} = \sum_{j=1}^{d} x_{ij}^{k,\pi}, i = 1, \ldots, n$, the row sums of $\mathbf{X}^{k,\pi}$, $k = 1, \ldots, n$, and by $\overline{s}^k$ their mean value, i.e.,

$$\overline{s}^k = \frac{\sum_{i=1}^{n} s_i^{k,\pi}}{n} = \frac{\sum_{i=1}^{n}\sum_{j=1}^{d} x_{ij}^{k,\pi}}{n} = \frac{\sum_{i=1}^{n}\sum_{j=1}^{d} x_{ij}^{k}}{n}.$$

Notice (from the last equality) that $\overline{s}^k$, the average value of the goods according to agent k, does not depend on the specific allocation π.

It is easy to see that a rearrangement $\pi \in \mathcal{A}$ unambiguously defines an admissible allocation κ_π via

$$j \in \kappa_\pi(i) \iff \pi_j(i) = 1. \tag{3}$$

Hence, the j-th item belongs to the i-th agent if and only if the j-th position of the i-th row in the matrices $\mathbf{X}^{k,\pi}$ takes value $x_{ij}^k > 0$ (and thus does not take value zero). For instance, the identity rearrangements $\pi_j(i) = i$, $i = 1, \ldots, n$, $j = 1, \ldots, d$, lead to the previously described allocation in which $\kappa(1) = \{1, \ldots, d\}$ and $\kappa(i) = \emptyset, i \neq 1$. For a given $\pi \in \mathcal{A}$, the elements in the i-th row of the matrix $\mathbf{X}^{k,\pi}$ represent the values given by the k-agent to the bundle $\kappa_\pi(i)$, received by the i-th agent. Hence, the utility gained by agent i – according to the valuation of agent k – is given by the $i-$th row sum of $\mathbf{X}^{k,\pi}$, i.e.,

$$u_k(\kappa_\pi(i)) = s_i^{k,\pi}.$$

In particular, the k-th row of the matrix $\mathbf{X}^{k,\pi}$ describes the bundle received by agent k, i.e., $s_k^{k,\pi} = u_k(\kappa_\pi(k))$ is the total utility the k-th agent gains for the goods that are allocated to the agent. Note that each column of $\mathbf{X}^{k,\pi}$ contains exactly one non-zero entry, coherently with the fact that each item is allocated to exactly one agent.

Example 2.1 (continued)**.** Consider the allocation $\kappa_\pi(1) = \{3, 4\}$, $\kappa_\pi(2) = \{1, 2\}$, i.e., the first agent receives the third and fourth item, whereas the second agent receives the other ones. The corresponding matrices $\mathbf{X}^{k,\pi}$ are given by

$$\mathbf{X}^{1,\pi} = \begin{pmatrix} 0 & 0 & 9 & 8 \\ 6 & 10 & 0 & 0 \end{pmatrix} \left| \begin{array}{l} s_1^{1,\pi} = 17 \\ s_2^{1,\pi} = 16 \end{array} \right. , \quad \mathbf{X}^{2,\pi} = \begin{pmatrix} 0 & 0 & 13 & 4 \\ 5 & 12 & 0 & 0 \end{pmatrix} \left| \begin{array}{l} s_1^{2,\pi} = 17 \\ s_2^{2,\pi} = 17 \end{array} \right. .$$

with rearrangement $\pi = (\pi_1, \pi_2, \pi_3, \pi_4)$ fully characterized by $\pi_1(2) = 1$, $\pi_2(2) = 1$, $\pi_3(1) = 1$, $\pi_4(1) = 1$ (see also (3)). Under this allocation, both agents have an utility equal to 17 ($s_1^{1,\pi} = s_2^{2,\pi} = 17$). The first agent believes (according to his own evaluation) that the other agent has

Electronic copy available at: https://ssrn.com/abstract=3512113

utility equal to 16 ($s_2^{1,\pi} = 16$) whereas the second agent believes the other agent has utility equal to 17 ($s_1^{2,\pi} = 17$).

3 Fairness criteria

The objective of fair allocation is to find an allocation that satisfies some optimality criteria that are generally referred to as fairness criteria. In the literature, there does not exist a single compelling definition of fairness, rather a broad variety of notions and rules have been introduced in economics and decision theory; see for instance Thomson (2011) and Lang and Rothe (2016) for surveys. Below, we discuss some fairness criteria that will be used in the remainder of the paper.

As we have seen in (3), each rearrangement $\pi \in \mathcal{A}$ corresponds to a unique allocation κ_π, and vice versa. In the remainder of the paper, we identify the notation κ_π with the corresponding rearrangement and use the same symbol π to refer to either a rearrangement or the corresponding allocation.

Minimum (social) inequality allocations. An intuitive criterion for assessing the fairness of an economic allocation is to measure the level of equality that it can ensure among agents. For a given agent k ($k = 1, \ldots, n$), we aim to obtain an allocation π that yields utilities $s_i^{k,\pi}$ that are as similar as possible; in this case, according to the valuation of the k-th agent, all other agents obtain approximately the same utility. Let $V_k(\pi)$ denote the variance of agents' utilities under the allocation π as measured by agent k, that is

$$V_k(\pi) = \frac{\sum_{i=1}^{n}(s_i^{k,\pi} - \overline{s}^k)^2}{n}.$$

To obtain an allocation that ensures minimum inequality among all agents, we thus aim to solve the optimization problem

$$V^* = \min_{\pi \in \mathcal{A}} V(\pi), \tag{4}$$

where the variance functional $V(\pi)$, defined as

$$V(\pi) = \frac{\sum_{k=1}^{n} V_k(\pi)}{n}, \tag{5}$$

measures the level of social inequality under the allocation π.

Since the number of possible allocations is finite, the minimum in (4) is attained and there always exists an allocation π^* satisfying $V(\pi^*) = V^*$. Any such π^* will be called a *minimum (social) inequality allocation*. In general, $V^* \geqslant 0$. In the particular case that $V^* = 0$, each agent believes that all agents experience exactly the same level of utility and we speak of *perfect social equality*. We note that the notion of social inequality is different from the notion of *equitability* (Brams et al. 2013). Under an equitable allocation, each agent assign the same utility to the bundle

Electronic copy available at: https://ssrn.com/abstract=3512113

received, i.e., $s_i^{i,\pi} = s_j^{j,\pi}$, $i, j \in \{1, \ldots, n\}$. When the total utility given by each agent to the d items is normalized to, say, 100, then a perfect social equality allocation is also equitable.

The notion of minimum inequality introduced above is novel. We will now establish its connection with other fairness criteria and provide evidence that a minimum inequality allocation performs very well also with respect to other well-established notions of fairness. We first recall these fairness notions and then explore their connection with minimum inequality allocations.

Envy-free allocations. One of the most widely used notions of fairness is that of *envy-freeness* (Foley 1967). An allocation π is said to be *envy-free* if no agent prefers the bundle received by any other agent to his own, that is, if

$$s_k^{k,\pi} \geqslant s_i^{k,\pi}, \text{ for all } k \in \{1, \ldots, n\} \text{ and } i \neq k.$$

An envy-free allocation is, for instance, obviously attained in the case in which all the $s_i^{k,\pi}$ are constant and thus equal to their average value $\overline{s}^k$; that is, when $V^* = 0$ (perfect social equality).

Lemma 3.1 (Sufficient condition for an envy-free allocation)**.** *If for all* $i, k \in \{1, \ldots, n\}$ *it holds that*

$$s_i^{k,\pi} = \overline{s}^k, \tag{6}$$

then the allocation π *is envy-free.*

Equivalently, if $V(\pi) = V^* = 0$, then π is an envy-free allocation. However, we stress that condition (6) is only sufficient to obtain an envy-free allocation, but not necessary. There might be envy-free allocations that do not satisfy (6), and, in general, an envy-free allocation might not even exist (consider the classical example of allocating one diamond and $(n-1)$ rocks). Therefore, it makes sense to look for allocations such that the envy among agents is as low as possible.

Minimum envy allocations. Following Lipton et al. (2004), we define the envy $e(\pi)$ of an allocation π as the maximal envy among any pair of agents, i.e.,

$$e(\pi) = \max\{e_{ij}(\pi); i, j \in \{1, \ldots, n\}\}, \tag{7}$$

where

$$e_{ij}(\pi) = \max\{0, s_j^{i,\pi} - s_i^{i,\pi}\}.$$

If $e_{ij}(\pi) = 0$, then agent i does not envy agent j. It is then immediate to observe that an allocation π is envy-free if and only if $e(\pi) = 0$. We say that π is a minimum envy allocation if $e(\pi)$ is minimum over the (finite) set of all possible allocations. Again, we note that a minimum envy allocation always exists.

Even if the functional $V(\pi)$ in (5) does not measure the fairness of an allocation in terms of

Electronic copy available at: https://ssrn.com/abstract=3512113

envy, we now show that the level of inequality $V^* = V(\pi^*)$ associated with a minimum inequality allocation π^* yields an upper bound for the minimum level of envy. To this end, denote by π^{**} a *minimum inequality allocation with minimum envy*, that is $V(\pi^{**}) = V^*$ and $e(\pi^{**}) \leqslant e(\pi^*)$ for all π^* satisfying $V(\pi^*) = V^*$. The proofs of the following propositions are relegated to the Appendix.

We first state a result for agents with *homogeneous preference*, i.e., for agents that assign the same values $x_1, \ldots, x_d$ to the d objects.

Proposition 3.2 (Bound for envy - homogeneous preferences)**.** *Let π^{**} be a minimum inequality allocation with minimum envy. If the n agents have homogeneous preferences, then it holds that*

$$e(\pi^{**}) \leqslant \sqrt{2nV^*}. \tag{8}$$

In particular, the envy $e(\pi)$ of a minimum envy allocation π is bounded by $\sqrt{2nV^*}$.

If one considers the general case of agents who may have *inhomogeneous* preferences, one still obtains an upper bound for the minimum envy in terms of V^*.

Proposition 3.3 (Bound for envy - inhomogeneous preferences)**.** *Let π^{**} be a minimum inequality allocation with minimum envy and let $V_{\min} = \min_{1 \leqslant k \leqslant n} V_k(\pi^{**})$ and $V_{\max} = \max_{1 \leqslant k \leqslant n} V_k(\pi^{**}) > 0$. It holds that*

$$e(\pi^{**}) \leqslant n\sqrt{\frac{2}{1+(n-1)C}\, V^*}, \tag{9}$$

where $C = \frac{V_{\min}}{V_{\max}}$.

Remark 3.1. We point out the following remarks.

(a) When $V^* = 0$, any π^* is an envy-free allocation. However, we stress that while an envy-free allocation might not exist (depending on the value matrix $\mathbf{X}$), an optimal allocation π^{**} with minimum envy always does. Since this minimum level of envy is bounded by $V^* = \min_{\pi \in \mathcal{A}} V(\pi)$, it thus makes sense to look for allocations that minimize $V(\pi)$.

(b) If $V_{\min} = 0$, Proposition 3.3 provides the rough bound $e(\pi^{**}) \leqslant n\sqrt{2\, V^*}$. In this case, in a similar way as in the proof of the proposition, one can still obtain the better bound

$$e(\pi^{**}) \leqslant n\sqrt{\frac{2}{1+(r-1)c}\, V^*},$$

where $r = \#\{k : V_k(\pi^{**}) > 0\}$, $c = \frac{v_{\min}}{V_{\max}}$, $v_{\min} = \min\{V_k(\pi^{**}) : V_k(\pi^{**}) > 0\}$.

(c) We conjecture the following sharp bound to hold in the general case of inhomogeneous preferences:

$$e(\pi^{**}) \leqslant n\sqrt{\frac{n}{2(n-1)} V^*};$$

Electronic copy available at: https://ssrn.com/abstract=3512113

see the Appendix for further details.

Proportional allocations. Another widely used notion of fairness is *proportionality*. First, for a given allocation π, we say that the $i-$th agent has a *fair share* if, according to the own valuation, the agent has received at least $1/n-$th of the total utility of the goods; formally, this is true if

$$s_i^{i,\pi} \geqslant \overline{s}^i.$$

If all agents receive a fair share, an allocation π is called *proportional*. It is immediate to note that that an envy-free allocation is also proportional (the converse is not true in general).

Maximum Nash welfare. An allocation π that maximizes the product of agents' utilities

$$U(\pi) = \prod_{i=1}^{n} s_i^{i,\pi}, \tag{10}$$

is called a maximum Nash welfare (MNW) allocation. Caragiannis et al. (2019) provides evidence that a MNW allocation performs well with respect to other fairness criteria. In particular, in Theorem 3.2 of the above paper, the authors show that a MNW allocation is envy-free up to one good.

4 Algorithms for fair allocation

Whereas there is a wide literature of algorithms for the case of cake-cutting problems (i.e., divisible items), for which we just mention Procaccia (2016), the fairly natural case of allocating indivisible goods has been somewhat neglected in its full generality, perhaps because of its more complicated nature.

In this section, we introduce two novel algorithms to treat numerically the problem of fair allocation without any restrictions on the number of agents/goods considered. These algorithms rely on the matrix formulation of the problem described in Section 2. In Section 5, we provide evidence that our methods are fast and yield high quality allocations, also in high-dimensional cases.

4.1 Minimum covariance algorithm (MinCov)

The first algorithm that we introduce is called the minimum covariance algorithm (MinCov). This algorithm is based on the observation that the n matrices $\mathbf{X}^k = (x_{ij}^k)$ defined in (1) satisfy

Electronic copy available at: https://ssrn.com/abstract=3512113

$$x_{ij}^k = \beta_j^k x_{ij}^1, \tag{11}$$

where $\beta_j^k = x_{1j}^k / x_{1j}^1$, $k = 1, \ldots, n, j = 1, \ldots, d$.

The MinCov algorithm uses rearrangements $\pi = (\pi_1, \ldots, \pi_d)$ having the property that there exist $j_1 \in \{1, \ldots, d\}$ such that $\pi_j(i) = i$ for $i \in \{1, \ldots, n\}$ and $j \neq j_1$. The matrices $\mathbf{X}^{k,\pi}$ are thus obtained from the matrices $\mathbf{X}^k$ by permuting (potentially all) elements in the j_1-th columns without affecting the others; we denote such rearrangements as $\pi(j_1)$.

Note that the variance functional in (5) can be written as:

$$V(\pi) = \frac{\sum_{k=1}^n V_k(\pi)}{n} = \frac{\sum_{k=1}^n \sum_{i=1}^n (s_i^{k,\pi} - \overline{s}^k)^2}{n^2} = \frac{\sum_{k=1}^n \left(\sum_{i=1}^n (s_i^{k,\pi})^2 - n(\overline{s}^k)^2 \right)}{n^2}.$$

In order to minimize $V(\pi)$, one has to minimize only the term in the last expression that depends on the rearrangement π chosen, that is $\sum_{k=1}^n \sum_{i=1}^n (s_i^{k,\pi})^2$. Defining, for some index $j_1 \in \{1, \ldots, d\}$, $s_{i(-j_1)}^{k,\pi} = \sum_{j \neq j_1} x_{ij}^{k,\pi}$, one has to minimize

$$\sum_{k=1}^n \sum_{i=1}^n (s_i^{k,\pi})^2 = \sum_{k=1}^n \sum_{i=1}^n \left(x_{ij_1}^{k,\pi} + s_{i(-j_1)}^{k,\pi} \right)^2 = \sum_{k=1}^n \sum_{i=1}^n \left[\left(x_{ij_1}^{k,\pi} \right)^2 + \left(s_{i(-j_1)}^{k,\pi} \right)^2 + 2 \left(x_{ij_1}^{k,\pi} \; s_{i(-j_1)}^{k,\pi} \right) \right].$$

If one uses rearrangements $\pi(j_1)$ that permute the j_1-th column only, one has to minimize the *covariance* term

$$C(\pi) = \sum_{k=1}^n \sum_{i=1}^n \left(x_{ij_1}^{k,\pi} \; s_{i(-j_1)}^{k,\pi} \right). \tag{12}$$

Since by assumption $x_{ij_1}^{k,\pi(j_1)} = \beta_{j_1}^k x_{ij_1}^{1,\pi(j_1)}$, it follows that $\pi(j_1)$ should be chosen such that

$$C(\pi) = \sum_{k=1}^n \sum_{i=1}^n \left(x_{ij_1}^{k,\pi(j_1)} \; s_{i(-j_1)}^{k,\pi(j_1)} \right) = \sum_{k=1}^n \sum_{i=1}^n \left(x_{ij_1}^{1,\pi(j_1)} \; \beta_{j_1}^k s_{i(-j_1)}^{k,\pi(j_1)} \right) = \sum_{i=1}^n \left(x_{ij_1}^{1,\pi(j_1)} \sum_{k=1}^n \beta_{j_1}^k s_{i(-j_1)}^{k,\pi(j_1)} \right)$$

is minimum. It is well-known from the Hardy-Littlewood inequality that the scalar product of two vectors is minimized if the vectors are oppositely ordered (Hardy et al. 1952, Th. 368). Hence, one should choose $\pi(j_1)$ such that $\mathbf{X}_{j_1}^{1,\pi(j_1)}$, the j_1-th column of $\mathbf{X}^{1,\pi(j_1)}$, becomes oppositely ordered to $\sum_{k=1}^n \beta_{j_1}^k \mathbf{S}_{(-j_1)}^{k,\pi(j_1)}$, where we set $\mathbf{S}_{(-j_1)}^{k,\pi} = (s_{1(-j_1)}^{k,\pi}, \ldots, s_{n(-j_1)}^{k,\pi})$. These considerations lead to the MinCov algorithm.

Electronic copy available at: https://ssrn.com/abstract=3512113

Minimum covariance algorithm (MinCov)

1. Let $\mathbf{X}^k$, $k = 1, \ldots, n$, be given $n \times d$ matrices and let n_{iter} be a positive integer.

2. Randomly select $j_1 \in \{1, \ldots, d\}$ and determine the rearrangement $\pi(j_1)$ such that the vector $\mathbf{X}_{j_1}^{1,\pi(j_1)}$ becomes oppositely ordered to $\sum_{k=1}^{n} \beta_{j_1}^k \mathbf{S}_{(-j_1)}^{k,\pi(j_1)}$. Ties are treated as *first coordinate comes first.*

3. Re-label $\mathbf{X}^{k,\pi(j_1)}$ as $\mathbf{X}^k$ and repeat Step 2. until n_{iter} iterations have been performed.

4. Output the matrix $\mathbf{X}^k$ so obtained and the corresponding final rearrangement π.

The vector $\mathbf{X}_{j_1}^{1,\pi(j_1)}$ contains only one non-zero element, at the row corresponding to the agent receiving the j_1-th item. As a consequence, oppositely ordering this vector essentially means re-allocating only item j_1. This is also how the algorithm is implemented in our code in order to avoid unnecessary sorting calculations.

At each iteration of MinCov, the inequality $V(\pi)$ of the system is strictly reduced unless the selected column of $\mathbf{X}^1$ is already properly ordered (in this latter case, there is no change). If the number n_{iter} of iterations is set large enough, the output rearrangement π can be expected to provide a value $V(\pi)$ close to the optimal one, V^*, and hence can be used to approximate a minimum inequality allocation π^*. The following example illustrates the different steps of the algorithm.

Example 4.1 (Illustration of MinCov)**.** As in Example 2.1, we consider the value matrix

$$\mathbf{X} = \begin{pmatrix} 6 & 10 & 9 & 8 \\ 5 & 12 & 13 & 3 \end{pmatrix},$$

with corresponding matrices $\mathbf{X}^1$ and $\mathbf{X}^2$ given as

$$\mathbf{X}^1 = \begin{pmatrix} 6 & 10 & 9 & 8 \\ 0 & 0 & 0 & 0 \end{pmatrix} \left| \begin{array}{l} s_1^1 = 33 \\ s_2^1 = \ \ 0 \end{array} \right. , \quad \mathbf{X}^2 = \begin{pmatrix} 5 & 12 & 13 & 4 \\ 0 & 0 & 0 & 0 \end{pmatrix} \left| \begin{array}{l} s_1^1 = 34 \\ s_2^1 = \ \ 0 \end{array} \right. .$$

We apply MinCov. For pedagogic reasons we do not randomly select $j_1 \in \{1, \ldots, d\}$ but we proceed sequentially.

Iteration 1: Take $j_1 = 1$ and let $\pi(0)$ be the identity rearrangement. Compute $\beta_1^1 = 1$, $\beta_1^2 = \frac{5}{6}$, and

$$\mathbf{S}_{(-1)}^{1,\pi(0)} = \begin{pmatrix} 27 \\ 0 \end{pmatrix}, \quad \mathbf{S}_{(-1)}^{2,\pi(0)} = \begin{pmatrix} 29 \\ 0 \end{pmatrix}, \quad \sum_{k=1}^{2} \beta_1^k \mathbf{S}_{(-1)}^{k,\pi(0)} = \begin{pmatrix} 27 + \frac{5}{6}29 \\ 0 \end{pmatrix}.$$

Change the order of elements in the first column of the matrices $\mathbf{X}^1$ and $\mathbf{X}^2$, i.e., choose $\pi(1)$ such

Electronic copy available at: https://ssrn.com/abstract=3512113

that the rearranged matrices $\mathbf{X}^{k,\pi(1)}$ are given by

$$\mathbf{X}^{1,\pi(1)} = \begin{pmatrix} 0 & 10 & 9 & 8 \\ 6 & 0 & 0 & 0 \end{pmatrix} \left| \begin{array}{l} s_1^{1,\pi(1)} = 27 \\ s_2^{1,\pi(1)} = \ \ 6 \end{array} \right. , \quad \mathbf{X}^{2,\pi(1)} = \begin{pmatrix} 0 & 12 & 13 & 4 \\ 5 & 0 & 0 & 0 \end{pmatrix} \left| \begin{array}{l} s_1^{2,\pi(1)} = 29 \\ s_2^{2,\pi(1)} = \ \ 5 \end{array} \right. .$$

Relabel $\mathbf{X}^{1,\pi(1)}$ as $\mathbf{X}^1$ and $\mathbf{X}^{2,\pi(1)}$ as $\mathbf{X}^2$.

Iteration 2: Take $j_1 = 2$. Compute $\beta_2^1 = 1$, $\beta_2^2 = \frac{6}{5}$, and

$$\mathbf{S}_{(-2)}^{1,\pi(0)} = \begin{pmatrix} 17 \\ 6 \end{pmatrix}, \quad \mathbf{S}_{(-2)}^{2,\pi(0)} = \begin{pmatrix} 17 \\ 5 \end{pmatrix}, \quad \sum_{k=1}^{2} \beta_2^k \mathbf{S}_{(-2)}^{k,\pi(0)} = \begin{pmatrix} 17 + \frac{6}{5}17 \\ 6 + \frac{6}{5}5 \end{pmatrix} = \begin{pmatrix} 37.4 \\ 12 \end{pmatrix}.$$

Hence, we choose $\pi(2)$ such that the rearranged matrices $\mathbf{X}^{k,\pi(2)}$ are given by

$$\mathbf{X}^{1,\pi(2)} = \begin{pmatrix} 0 & 0 & 9 & 8 \\ 6 & 10 & 0 & 0 \end{pmatrix} \left| \begin{array}{l} s_1^{1,\pi(2)} = 17 \\ s_2^{1,\pi(2)} = 16 \end{array} \right. , \quad \mathbf{X}^{2,\pi(2)} = \begin{pmatrix} 0 & 0 & 13 & 4 \\ 5 & 12 & 0 & 0 \end{pmatrix} \left| \begin{array}{l} s_1^{2,\pi(2)} = 17 \\ s_2^{2,\pi(2)} = 17 \end{array} \right. .$$

Relabel $\mathbf{X}^{1,\pi(2)}$ as $\mathbf{X}^1$ and $\mathbf{X}^{2,\pi(2)}$ as $\mathbf{X}^2$.

At this point one can check that further iterations will not lead to a change in the output matrices $\mathbf{X}^1$ and $\mathbf{X}^2$. The bundle each agent receives with associated values can be observed from the first row of the first matrix and the second row of the second matrix, respectively. In particular, the final allocation κ is given by $\kappa(1) = \{3, 4\}$ and $\kappa(2) = \{1, 2\}$. Observe that the final allocation satisfies minimum inequality and is equitable. However, perfect social equality cannot be achieved in this case.

Finally, note that the MinCov algorithm is applicable whenever the matrices $\mathbf{X}^k$ meet condition (11), which is in particular satisfied in the fair allocation problem we consider.

4.2 Minimum covariance algorithm with target (MinCovTarg)

The MinCov algorithm can be used to find approximate minimum inequality allocations π^*. Since by Propositions 3.2–3.3, the level of envy of such rearrangements is bounded by the total inequality $V(\pi^*)$ present in the economic system, one could expect π^* to perform reasonably well when an envy-based fairness criterion is used.

However, there might still be room for improvement when considering a specific fairness criterion such as envy-freeness or minimum envy. In fact, for any minimum inequality allocation π^* there are at least $n!$ other minimum inequality allocations π with $V(\pi) = V^*$ obtained by permuting the rows in the corresponding rearranged matrices $\mathbf{X}^{k,\pi^*}$(without changing the value of V^*). In other words, one could simply shuffle the n bundles received by the agents under π^*: each shuffle affects the list of goods allocated to each agent, and thus the corresponding subjective utility, but not the total inequality of the economic system. Hence, we are missing a rule to determine which bundle

Electronic copy available at: https://ssrn.com/abstract=3512113

is allocated to each agent, which does not matter when using inequality as fairness criterion, but does so for other fairness criteria.

On these grounds, we propose an extension of MinCov, named minimum covariance algorithm with target (MinCovTarget) which is able to find *envy-free* allocations, if they exist, or at least allocations with minimum envy.

Minimum covariance algorithm with target (MinCovTarget)

1. Let $\mathbf{X}^k$, $k = 1, \ldots, n$, be given $n \times d$ matrices and let n_{iter} be a positive integer.
2. Define the matrices $\mathbf{Y}^k, k = 1, \ldots, n$, as the matrices $\mathbf{X}^k$ with an additional column t_k that contains only zeros, except on the k-th row, where the negative of a target value τ_k is set.
3. Run MinCov for n_{iter} iterations on the matrices $\mathbf{Y}^k$, with the exception that the columns t_k are never selected for rearrangement.
4. Remove from the obtained matrices the last column in order to obtain the output matrices $\mathbf{X}^k$ and the corresponding output rearrangement π.

Since MinCov produces matrices with row sums that are as equal as possible, it follows that, by adding the extra column with negative values, the output matrices $\mathbf{X}^k$, $k = 1, \ldots, n$, will show a tendency to have a larger k-th row sum (i.e., the utility of the k-th agent, according to the own valuation, is greater than the utility of all others). Hence, MinCovTarget offers a trade-off between the criteria of social inequality and envy. The target values τ_k must be calibrated depending on the initial value matrix $\mathbf{X}$. Choosing (too) large target values results in individually better allocations but would neglect the fairness criteria. For agent valuations that are standardized or not too inhomogeneous, one can choose a single target value $\tau_k = \tau, k = 1, \ldots, n$. If $\tau_k = 0, k = 1, \ldots, n$, MinCovTarget reduces to MinCov and proxies a minimum inequality allocation. Note that the negative target values in MinCovTarget are just artificial variables used to modify the variance function and do not represent goods to be allocated to the agents.

4.3 Maximum Nash welfare algorithm (MNW)

For the sake of comparison, we introduce an auxiliary algorithm to show how our two main algorithms behave with respect to maximum Nash welfare . Finding MNW allocations is computationally hard and only approximate solutions can be obtained. To obtain such approximation, we proceed with a so-called local trades approach. This approach consists in applying a sequence of random local trades between two agents: the single trade is performed only if it improves the Nash welfare objective. This leads to the following algorithm that we refer to as the maximum Nash welfare algorithm (MNW).

Electronic copy available at: https://ssrn.com/abstract=3512113

Maximum Nash welfare algorithm (MNW)

1. Randomly allocate each item to one of the n agents and let n_{iter} be a positive integer.
2. Randomly select $j \in \{1, \ldots, d\}$ and $i \in \{1, \ldots, n\}$. Allocate item j to agent i only if it strictly increases the utility $U(\pi)$ in (10); otherwise do nothing.
3. Repeat Step 2. until n_{iter} iterations have been performed.
4. Output the allocation π so obtained.

Note that this algorithm is different from the (yet approximate) algorithm for the MNW problem developed in Caragiannis et al. (2019). There, the utility objective (10) is first approximated and then mixed integer linear programming (MILP) is used to obtain an approximation for the MNW allocation.

4.4 Local trades algorithm

The same local trades approach can be used to minimize social inequality or envy. To distinguish its application to different objective functions in what follows, we refer to this approach as the *Local trades algorithm* (LocalTrades).

Local trades algorithm (LocalTrades)

1. Randomly allocate each item to one of the n agents and let n_{iter} be a positive integer.
2. Randomly select $j \in \{1, \ldots, d\}$ and $i \in \{1, \ldots, n\}$. Allocate item j to agent i only if this strictly decreases the variance $V(\pi)$ in (5) (alternatively, the envy $e(\pi)$ in (7)) of the allocation so obtained; otherwise do nothing.
3. Repeat Step 2. until n_{iter} iterations have been performed.
4. Output the allocation π so obtained.

5 Numerical study

In this section, we apply the four algorithms introduced so far to find fair allocations under the various fairness notions considered. For the sake of comparison, in all our numerical experiments we fix the number of iterations n_{iter} for each algorithm, and we also run each algorithm until no improvement is observed for a given number of iterations, which is set equal to the number d of items to be allocated. In this latter case we say that the algorithm has *converged*.

Electronic copy available at: https://ssrn.com/abstract=3512113

Here it is important to remark again that the problem of allocation for the rules considered is $\mathcal{NP}$-hard, and one cannot expect our algorithms to terminate by finding a global solution, but rather a local one. Moreover, the algorithms differ substantially with respect to the number of iterations that are needed to satisfy the stopping rule of convergence. MinCov and MinCovTarget converge very quickly whereas MNW and LocalTrades require a higher number of iterations. These considerations seem to indicate that our approaches are more suitable to treat high-dimensional cases, where approximate MILP approaches like in Caragiannis et al. (2019) might be computationally more involved. Finally note that in all our (high-dimensional) experiments convergence was always reached within 10^6 iterations.

Set-up. We consider that the number d of items to be allocated is a multiple of the number of agents, i.e., $d = kn$, $k \in \mathbb{N}$. This is done just for illustrative reasons as our algorithms do not have restrictions on the (possibly high) number of items/agents to be used. The values $x_{1j}, j = 1, \ldots, d$, assigned by the first agent to the d items are drawn independently from a uniform distribution $U(0, 100)$. The utilities of the other $(n-1)$ agents are also randomly drawn as

$$x_{ij} \in U[x_{1j}(1-\varepsilon), x_{1j}(1+\varepsilon)], \quad i = 2, \ldots, n, \quad j = 1, \ldots, d, \tag{13}$$

for some fixed $\varepsilon > 0$. In what follows, we use $\varepsilon = 0.5$, but different values for ε yield similar results. Next, we slightly adjust the simulated values of agent utilities so that the total utility given by each agent to the d items is normalized to 100 and an envy-free allocation is guaranteed to exist. To ensure that the envy-free solution cannot be directly observed from $\mathbf{X}$ we randomize the order of columns herein. Finally, we allocate the items randomly to the different agents to obtain an initial configuration and apply the various algorithms. We repeat this procedure 1000 times. For the algorithm MinCovTarget, we set a single target value $\tau_k = 20, k = 1, \ldots, n$.

5.1 Minimum social inequality

Table 1 provides detailed statistics about the social inequality of the final allocation found by the various algorithms. Note that under the initial (random) allocation each agent receives on average d/n items. As each item received has an expected value of $100/d$, it follows that under the initial allocation the variance $V(\pi)$ is proportional to $1/(nd)$, which is consistent with the pattern observed in the table.

Table 1 shows that LocalTrades yields reasonably good solutions, but MinCov is outperforming and yields solutions with very low variance (inequality). Starting from a random allocation of the different goods, MinCov is able to reduce social inequality by a factor of 10. The performance of MNW and MinCovTarget, which are not specifically designed to yield low values for the variance, is poor with MNW showing the worst performance for this criterion.

The outperformance of MinCov becomes evident in Figure 1, which shows the percentage of

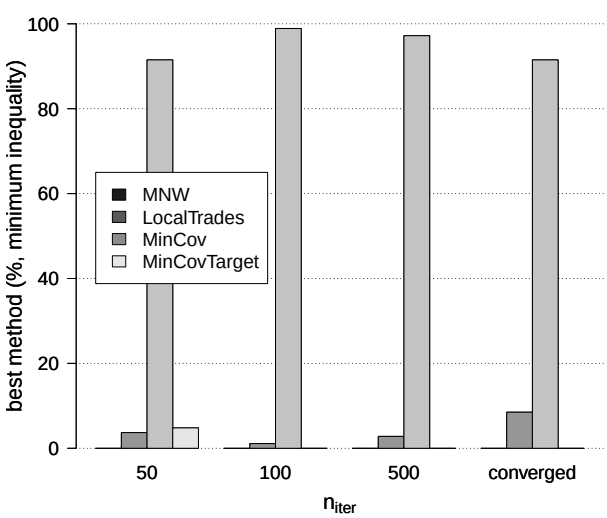
Electronic copy available at: https://ssrn.com/abstract=3512113

times that a given algorithm performs best for the minimum inequality criterion. The figure includes the cases $(n, d) = (10, 100)$ and $(n, d) = (100, 1000)$. For almost all 1000 initial configurations, MinCov provides the fairest allocation.

	min.	med.	max.	mean	min.	med.	max.	mean
Panel A: $d = 1000$		$n = 10$				$n = 100$		
Initial	0.87	1.28	1.82	1.31	0.12	0.14	0.16	0.14
MNW	2.64	7.06	7.95	6.66	0.02	0.04	0.05	0.04
LocalTrades	0.00	0.00	0.02	0.01	0.00	0.01	0.02	0.01
MinCov	0.00	0.00	0.02	0.00	0.00	0.01	0.01	0.01
MinCovTarget	2.78	6.53	8.34	6.09	0.00	0.02	0.03	0.02
Panel B: $d = 100$		$n = 10$				$n = 25$		
Initial	8.47	12.55	17.76	12.76	4.17	5.43	7.06	5.47
MNW	4.39	8.63	13.12	8.69	1.28	1.99	2.93	2.01
LocalTrades	0.35	0.88	1.80	0.94	0.24	0.60	1.03	0.61
MinCov	0.24	0.63	1.35	0.68	0.19	0.51	0.87	0.52
MinCovTarget	2.32	6.23	11.36	6.42	0.33	0.93	2.15	1.01

Table 1: We compute the n variances of the row sums of n matrices for the initial configuration as well as after application of the various algorithms. The algorithms are run until there are no improvements in objective value for d consecutive updates. We store the minimum, median, maximum, and average variance (i.e., social inequality). Each shown statistic represents the average over 1000 replications of the individual summary statistics.

(a) Minimum inequality obj. – $n = 10$, $d = 100$.



(b) Minimum inequality obj. – $n = 100$, $d = 1000$.

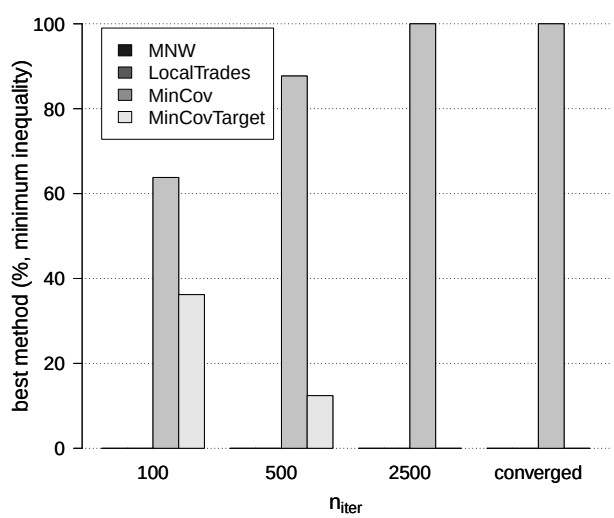


Figure 1: Best method for finding minimum inequality allocations, for different combinations of (n, d), and number of iterations used. Rates of success against minimum inequality are computed over 1000 different random initial configurations. LocalTrades is here applied to the minimum inequality objective function.

Electronic copy available at: https://ssrn.com/abstract=3512113

5.2 Minimum envy

Similarly as above, Figure 2 shows the percentage of times that one of the four algorithms used performs best when minimum envy is used as a fairness rule. It is evident that MinCovTarget is the champion algorithm in providing the best allocation, with MNW being a close competitor.

Indeed, we found a tie most of the times when both methods turned out to find an allocation with zero envy. This is also confirmed by a detailed inspection of the simulation results, which reveals that for the case $(n, d) = (10, 100)$, MinCovTarget finds in 97.5% of the simulations a no-envy solution, whereas MNW in 95.5% of the cases. When $(n, d) = (100, 1000)$, we have a no-envy solution in 85.0% of the simulations for MinCovTarget and 76.9% for MNW, indicating that the outperformance of MinCovTarget is increasing with dimensionality. In the case $(n, d) = (100, 1000)$, it is noteworthy to mention that in 65.7% of the cases *both* methods find a no-envy solution and thus that in 96.2% of the cases at least one of the methods find a no-envy solution, suggesting that it appears useful to run both MNW and MinCovTarget for minimizing envy.

(a) Minimum envy obj. – $n = 10$, $d = 100$.

(b) Minimum envy obj. – $n = 100$, $d = 1000$.

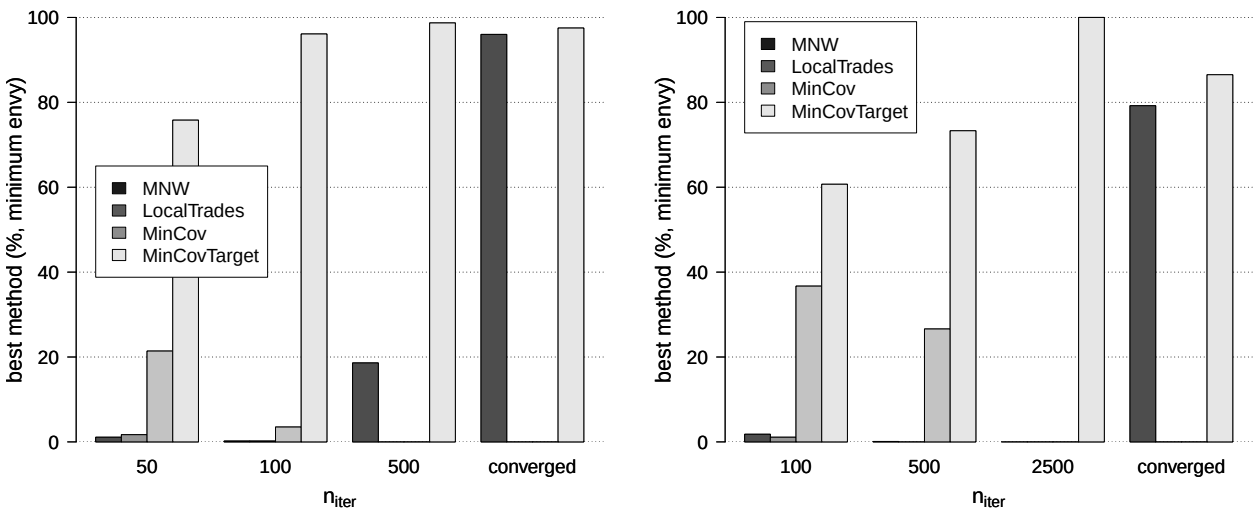


Figure 2: Best method for finding minimum envy allocations, for different combinations of (n, d), and different number of iterations used. Rates of success against minimum envy are computed over 1000 different random initial configurations. LocalTrades is here applied to the envy objective function.

Figure 3 shows for each algorithm, how many agents on average have received a fair share, as well as how many believe they have received the best bundle (i.e., that they obtain goods with a total utility that is greater than that for all other agents). Note that an allocation is envy-free if and only if all agents believe to have received the best bundle. From Figure 3, we observe that both MinCovTarget and MNW generally find a proportional allocation (i.e., providing a fair share to all the agents) and provide all agents with the best deal after a sufficient number of iterations

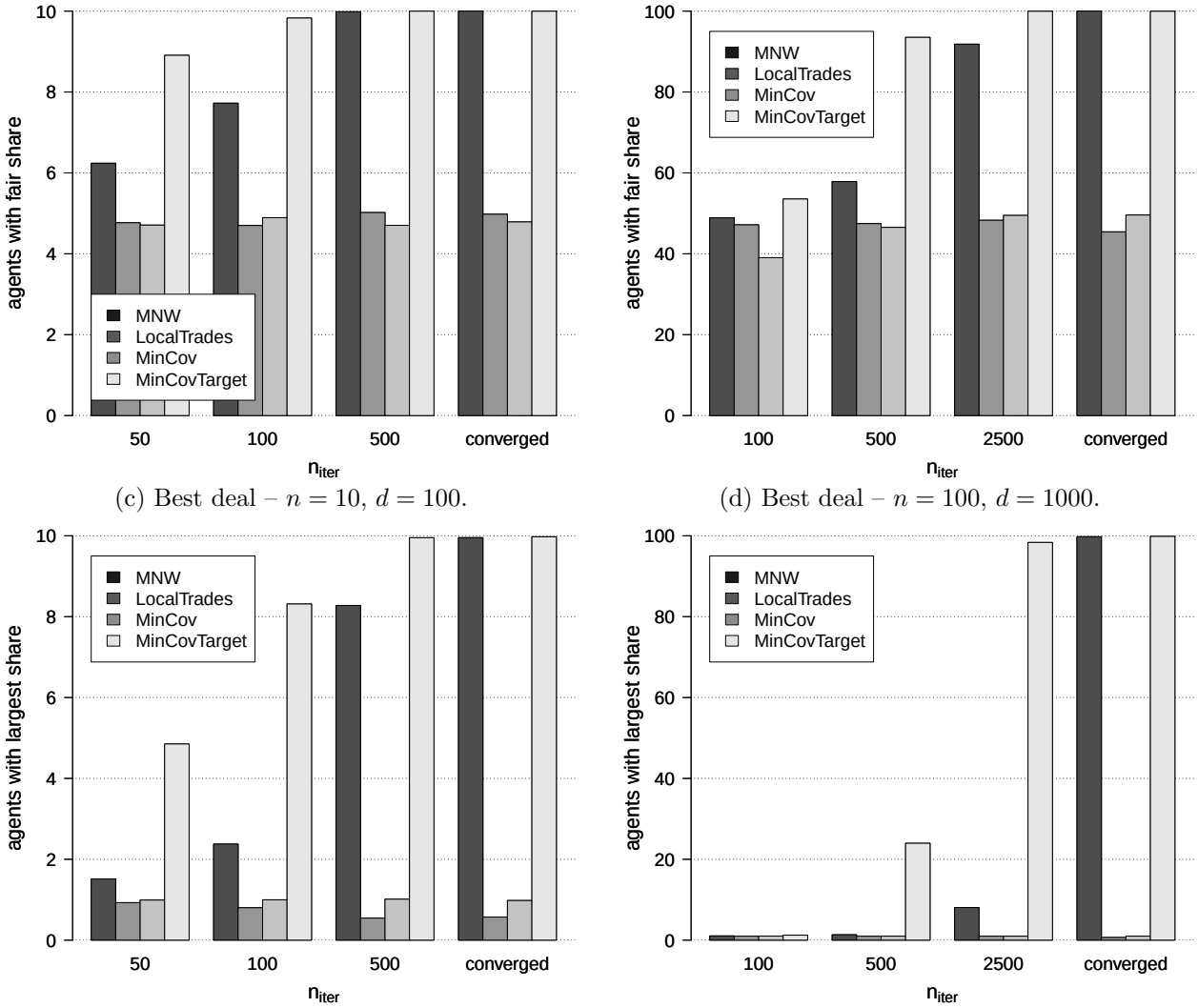
Electronic copy available at: https://ssrn.com/abstract=3512113

(recall from Figure 2 and its discussion that both MNW and MinCovTarget often find envy-free allocations). Inspection of the simulation results reveals that, when $(n, d) = (100, 1000)$, in 99.2% of the cases 99 or 100 agents receive a best deal (for both MNW and MinCovTarget). Note that MinCovTarget however outperforms MNW with regards to the number of cases it provides a best deal to all 100 agents (85.0% versus 76.9%).

With respect to minimal envy and best deal, MinCov and LocalTrades show similar results but are clearly outperformed by MNW, and in particular by MinCovTarget.

(a) Fair share – $n = 10$, $d = 100$.

(b) Fair share – $n = 100$, $d = 1000$.



(c) Best deal – $n = 10$, $d = 100$.

(d) Best deal – $n = 100$, $d = 1000$.

Figure 3: Average number of agents with fair share and number of agents with largest share for different combinations of (n, d), and different number of iterations used. Averages are computed over 1000 different random initial configurations. LocalTrades is here applied to the envy objective function.

A natural question is how sensitive the MinCovTarget algorithm is when varying the target

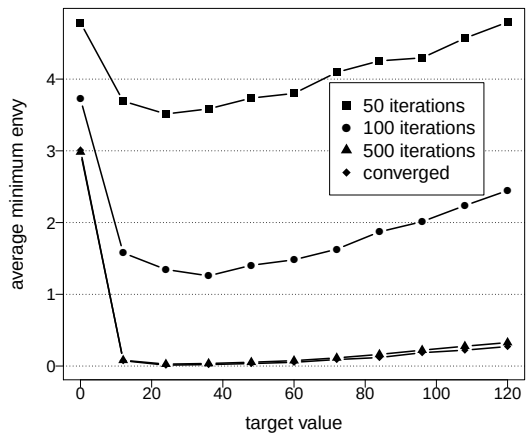
Electronic copy available at: https://ssrn.com/abstract=3512113

value, which in our study was set equal to 20. Figure 4 displays the envy $e(\pi)$ (left part of the Figure) as well the level of social inequality $V(\pi)$ (right) of the output allocation π produced when varying the target value between 0 and 120. As expected, the use of a positive target values is required to lower the envy in the system.

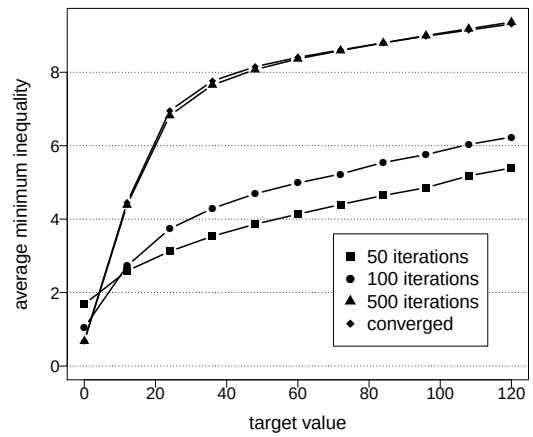
In practice, one needs to calibrate the target values in a range where they are not too small to select allocations with minimum inequality but possibly high envy, and not too large to prioritize no-envy over minimal inequality allocations. A satisfying observation in this respect is that the MinCovTarget algorithm is robust as there is a wide range of target values for which a low level of envy can be accompanied by a relatively low value of inequality.

Since running the MinCovTarget algorithm is fast, one could always produce similar plots as in Figure 4 to calibrate the value(s) of the target variable to the specific case under study.

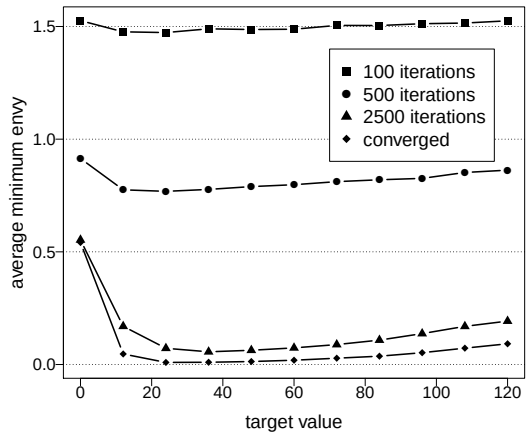
(a) Envy obj. – $n = 10$, $d = 100$.



(b) Inequality obj. – $n = 10$, $d = 100$.



(c) Envy obj. – $n = 100$, $d = 1000$.



(d) Inequality obj. – $n = 100$, $d = 1000$.

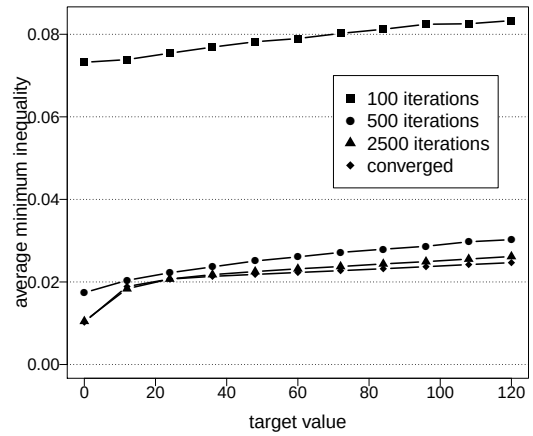


Figure 4: Average values of the minimum inequality functional $V(\pi)$ and of the envy attained after application of MinCovTarget versus the choice of a homogeneous target value, for different values of (n, d) and number of iterations. Averages are computed over 1000 different random initial configurations.

Electronic copy available at: https://ssrn.com/abstract=3512113

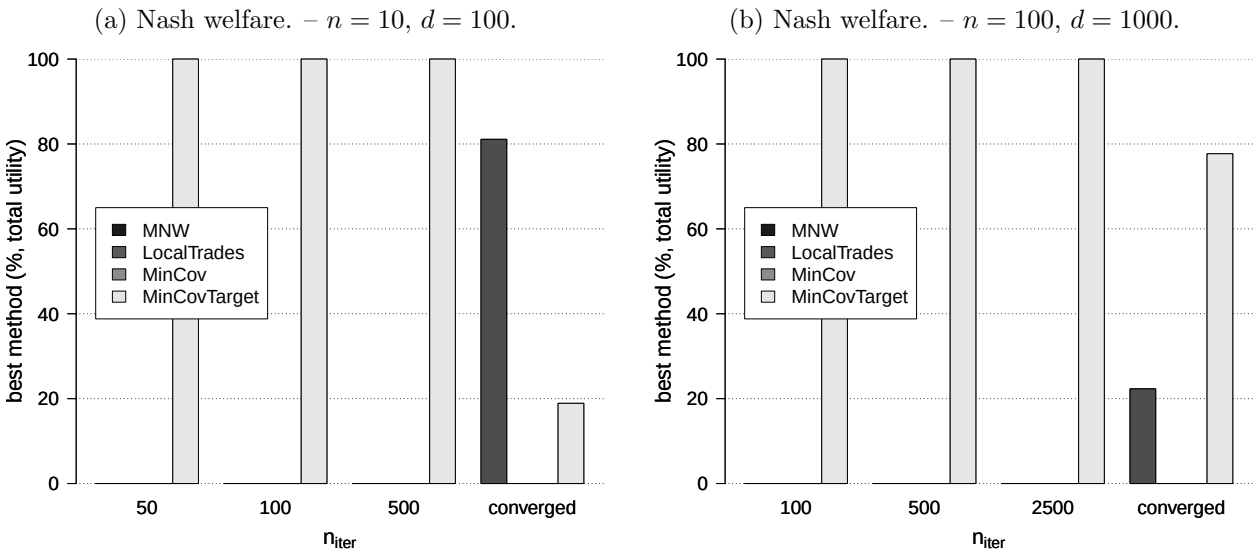


Figure 5: Best method for finding MNW allocations, for different combinations of (n, d), and different number of iterations used. Rates of success against MNW are computed over 1000 different random initial configurations. LocalTrades is here applied to the envy objective function.

5.3 Maximum Nash welfare

In this section, we also compare the value of Nash welfare produced by our algorithms. We found that for the case $(n, d) = (10, 100)$, MNW yields on average a value of 28.4 for the (log) Nash welfare $\log U(\pi)$; MinCovTarget is a close second-best with an average value equal to 28.2. The other algorithms perform less well. For the case $(n, d) = (100, 1000)$, MinCovTarget yields even slightly higher values for $\log U(\pi)$ than MNW. These observations are also consistent with Figure 5, where one can see that for higher dimensional cases MinCovTarget find allocations with higher Nash welfare.

Again, this seems to suggest that for high-dimensional setting as one implied for instance by the allocation of different units of medicine/vaccines to various medial divisions/hospitals, MinCovTarget could provide an alternative to algorithms aiming at maximizing Nash welfare.

Interestingly enough, the analysis of the Nash welfare also allows us to conclude that the MinCovTarget algorithm yields solutions that exhibit Pareto optimality. To see why this is true, note that the property of Pareto optimality is guaranteed for MNW solutions (as they solve for a product of utilities). For the case of MinCovTarget Pareto optimality is only guaranteed when the values of the target parameters τ_k, $k = 1, \ldots, n$, are set high enough, as can be seen directly from the definition. As both methods lead to similar values for the Nash welfare, we can conclude that in our simulations MinCovTarget leads approximately to Pareto optimal solutions also for moderate values of the target.

As the different algorithms require limited (permanent) memory usage their practical limits are

Electronic copy available at: https://ssrn.com/abstract=3512113

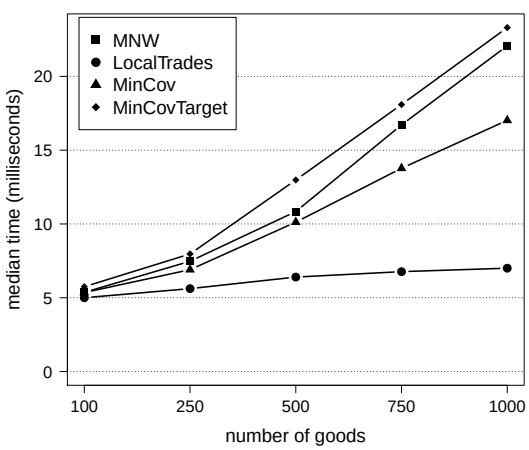
mainly driven by computational speed. Figure 6 shows how the computation times of the different algorithms scale with the number of items and agents.

In Figure 6a, we fix the number of agents at $n = 10$ and vary the number of items from $d = 100$ to $d = 1000$. We observe that for the different algorithms the number of goods has a negligible impact on the time per iteration but affects the number of iterations to obtain convergence. We find that the total computation time to convergence is approximately linear in the number of goods.

In Figure 6b, we vary the number of agents but keep the number of items fixed at $d = 1000$. In this case, we observe quadratic growth in time for all algorithms (MNW apart, for which it remains linear) since at each iteration the envy between all pairs of agents needs to be computed.

The figure shows that even in case of large number of items/agents, final allocations are obtained within a second.

(a) Varying number of goods.



(b) Varying number of agents.

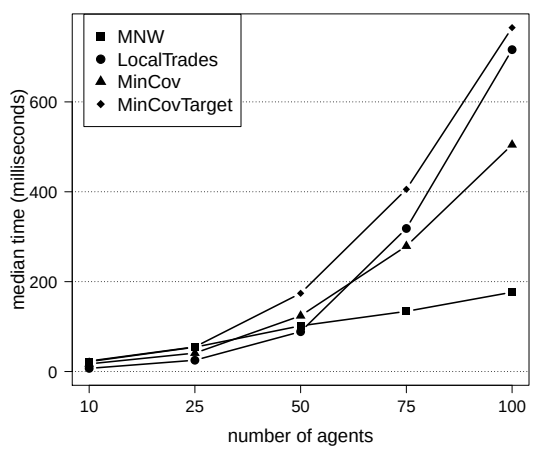


Figure 6: Computation times to convergence of the various algorithms designed for fair allocation.

6 Conclusions

This paper introduces a new mathematical framework for the fair allocation of indivisible goods. We show that this classical problem can be cast as the problem of finding an optimal column rearrangement of multiple matrices. This formulation leads to the design of two new algorithms (MinCov and MinCovTarget) that are suitable to deal with the novel notion of social inequality, but also produces promising results with respect widely used criteria like minimum envy, proportionality and Nash welfare.

In our simulations, MinCov provides allocations that reduce social inequality by an order of magnitude if compared to a random assignment, whereas MinCovTarget yields excellent results in finding envy-free allocations (envy-free allocations are obtained in almost all cases) or allocations with minimum envy. Numerical results show that MinCovTarget allocations are at least as good

Electronic copy available at: https://ssrn.com/abstract=3512113

as (and especially in higher dimensions outperforming) MNW allocations with respect to minimal envy and Nash welfare.

We remark that the problems at hand are $\mathcal{NP}$-hard, hence finding the global optimum is out of reach unless for low-dimensional instances where linear programming or full enumeration (i.e., considering all possible rearranged matrices) can be performed. As a consequence one can always provide examples (especially low-dimensional ones) under which the algorithms stop at a local minimum different from the global one, or where different random paths lead to substantially different results. However, for moderately high-dimensional problems where exact techniques cannot be applied, numerical illustrations show that our algorithms provide excellent results for a variety of different fairness rules and utility criteria.

In conclusion, the MinCov and MinCovTarget algorithms are straightforward to implement and are able to deal with the fair allocation problem for an arbitrary number of agents/objects. For instance, when allocating 1000 items among 100 agents the algorithms find (approximate) optimal allocations within less than a second. They can e.g. be used to fairly allocate scarce medical supplies among hospitals. In a first step, each hospital could assign a utility value to the various medical items. Next, depending on the fairness criterion used, MinCov or MinCovTarget can be ran to optimally allocate the items among all hospitals.

Acknowledgments We thank three anonymous reviewers for their very helpful suggestions on a previous version of this paper.

A Proofs of Propositions 3.2–3.3

The proofs of Propositions 3.2–3.3 require the following lemma.

Lemma A.1. *Let $x_1, x_2, \ldots, x_n$, be n real numbers in increasing order having mean $\overline{x} = \sum_{i=1}^n x_i/n$. For $\xi > 0$, let*

$$\mathcal{R} = \{j \in \{1, \ldots, n-1\};\ x_n - x_j \geqslant \xi\}.$$

If $\#\mathcal{R} = r$, with $1 \leqslant r \leqslant n-1$, then

$$V(x_1, \ldots, x_n) = \frac{\sum_{i=1}^n (x_i - \overline{x})^2}{n} \geqslant \frac{r}{n(r+1)}\xi^2.$$

Proof of Lemma A.1. First, note that

$$\frac{\sum_{i=1}^n (x_i - \overline{x})^2}{n} = \frac{1}{n^2}\sum_{i<j}(x_i - x_j)^2.$$

We can assume that $x_j = x_n - \xi, 1 \leqslant j \leqslant r$, otherwise the value of $V(x_1, \ldots, x_n)$ would be larger.

Electronic copy available at: https://ssrn.com/abstract=3512113

Hence, for fixed $x_n \in \mathbb{R}$, consider the problem

$$\min_{x_{r+1},\dots,x_{n-1}\in\mathbb{R}} \frac{1}{n^2}\sum_{i<j}(x_i - x_j)^2.$$

It is easy to verify that the solution of the above minimum is attained by the only point satisfying first order conditions; that is, $x^*_{r+1} = \cdots = x^*_{n-1} = x_n - \frac{r}{r+1}\xi$. Under the assumptions of the Lemma, we then obtain

$$\frac{\sum_{i=1}^n (x_i - \overline{x})^2}{n} \geqslant \min_{x_{r+1},\dots,x_n\in\mathbb{R}} \frac{1}{n^2}\sum_{i<j}(x_i - x_j)^2 = \frac{r}{n(r+1)}\xi^2.$$

□

A.1 Proof of Propositions 3.2

Assume that $e(\pi^{**}) > 0$ (otherwise (8) trivially holds) and that it is attained by agent i envying the bundle received by agent $j \neq i$, i.e.,

$$e(\pi^{**}) = s_j^{i,\pi^{**}} - s_i^{i,\pi^{**}} > 0.$$

Given the agents' homogeneous preferences, for each agent $k \in \{1,\dots,n\}$ we have that

$$s_j^{k,\pi^{**}} - s_i^{k,\pi^{**}} = e(\pi^{**}). \tag{14}$$

Using Lemma A.1 with $r = 1$, it follows that

$$V_k(\pi^{**}) \geqslant \frac{e(\pi^{**})^2}{2n}, k \in \{1,\dots,n\}$$

and

$$V^* = V(\pi^{**}) = \frac{\sum_{k=1}^n V_k(\pi^{**})}{n} \geqslant \frac{e(\pi^{**})^2}{2n},$$

from which the inequality (8) follows.

We conclude the proof by showing that the upper bound in (8) is *sharp* in the sense that, in general, it *cannot* be lowered for agents with homogeneous preferences. Indeed, consider $d = n - 1$ items to be allocated among n agents. The first item is valued (by everyone) M, the others $M/2$. The allocation with minimum V prescribes that each item is allocated to one of the agents, with a single agent – say agent 1 – not receiving anything. In this case one finds $V^* = M^2/(2n)$ and that the maximal envy attained by agent 1 (because one of the other agents received the first item) is $e(\pi^{**}) = M = \sqrt{2nV^*}$.

Electronic copy available at: https://ssrn.com/abstract=3512113

A.2 Proof of Propositions 3.3

Assume that $e(\pi^{**}) > 0$ (otherwise (9) trivially holds) and that it is attained by agent i envying the bundle received by agent $j \neq i$. Let $V_{\min} = \min_{1 \leqslant k \leqslant n} V_k(\pi^{**})$, $V_{\max} = \max_{1 \leqslant k \leqslant n} V_k(\pi^{**}) > 0$, and $C = V_{\min}/V_{\max}$. In a similar way as in the proof of Proposition A.1, it follows that

$$V_i(\pi^{**}) \geqslant \frac{e(\pi^{**})^2}{2n}. \tag{15}$$

Note that, for $j \neq i$, one has

$$V_j(\pi^{**}) \geqslant V_{\min} = C\, V_{\max} \geqslant C\, V_i(\pi^{**}). \tag{16}$$

Using (15) and (16), one obtains

$$V^* = \frac{\sum_{k=1}^{n} V_k(\pi^{**})}{n} \geqslant \frac{(1+(n-1)C)}{n}\, V_i(\pi^{**}) \geqslant \frac{e(\pi^{**})^2}{2n}\, \frac{(1+(n-1)C)}{n},$$

which is (9).

A.3 Conjecture in Remark 3.1(c)

Let π^{**} represent a minimum inequality allocation with minimum envy. Then, we conjecture that

$$e(\pi^{**}) \leqslant n\sqrt{\frac{n}{2(n-1)} V^*}. \tag{17}$$

Assume that $e(\pi^{**}) > 0$ (otherwise (17) trivially holds) and partition the set of agents into the two subsets

$$\overline{\mathcal{R}} = \{k \in \{1, \ldots, n\} : |s_i^{k,\pi^{**}} - s_j^{k,\pi^{**}}| < e(\pi^{**}), \text{ for all } i \neq j\},$$

and $\mathcal{R} = \{1, \ldots, n\} \setminus \overline{\mathcal{R}}$. Also, let $r = \#\mathcal{R}$. Bundles evaluations of agents in $\overline{\mathcal{R}}$ differ by less than $e(\pi^{**})$; thus, only agents belonging to $\mathcal{R}$ can have an envy of at least $e(\pi^{**})$, and hence $r \geqslant 1$. If $r = 1$, with $\mathcal{R} = \{k\}$ one could swap the bundle received by agent k with the bundle of the envied agent, canceling the envy of the first one, leaving the variance unchanged, and decreasing the envy of the allocation. Hence, $2 \leqslant r \leqslant n$.

Now consider the matrix $S = (s_j^{i,\pi^{**}})$, where the element (i, j) is the valuation by agent i of the bundle received by agent j under the allocation π^{**}. Re-labelling the agents, we can always write

$$S = \begin{pmatrix} S_{\mathcal{R}\,\mathcal{R}} & S_{\mathcal{R}\,\overline{\mathcal{R}}} \\ S_{\overline{\mathcal{R}}\,\mathcal{R}} & S_{\overline{\mathcal{R}}\,\overline{\mathcal{R}}} \end{pmatrix},$$

so that the rows/columns of agents in $\mathcal{R}$ appear first. Suppose that we put a cross on cell (i, j) to

Electronic copy available at: https://ssrn.com/abstract=3512113

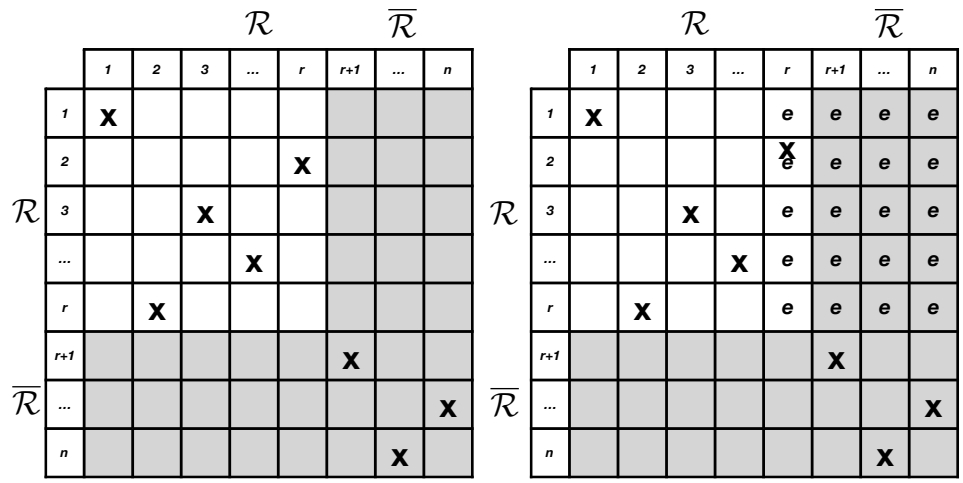


Figure 7

mean that agent i receives the bundle j. The allocation π^{**} is then illustrated by putting a cross on the main diagonal of the matrix S. We have already noted above that any allocation derived from π^{**} by a permutation of the bundles is still optimal and, by definition of π^{**}, must have at least the same envy. Any such allocation can be illustrated by putting exactly one cross in each column and in each row of S; see Figure 7, left. There are $n!$ such allocations.

Suppose also that we we label with e those bundles that, if assigned to agent i (i.e., if marked with a cross) will result in envy of at least $e(\pi^{**})$. Formally, cell (i, j) is marked with an an e when

$$\max_{k \neq j} \left(s_k^{i,\pi^{**}} - s_j^{i,\pi^{**}} \right) \geqslant e(\pi^{**}). \tag{18}$$

By definition of $\overline{\mathcal{R}}$, such e's cannot appear in the last $n - r$ rows of matrix S. On the other hand, each of the first r rows must have at least one e. Since we can always put $(n - r)$ crosses on the last $(n - r)$ rows with no e's, one needs $n - r$ columns of e plus an additional column of e to force one cross to appear over an e in the first r rows. Summarizing, if one requires π^{**} to be an optimal allocation with minimum envy equal to $e(\pi^{**})$, we conjecture that the least number of e that can be used is illustrated in Figure 7, right, where the last $(n - r + 1)$ columns contain an e up to the r-th row. It is immediate to see that if one removes just one e from this configuration, then one can re-allocate the bundles to obtain zero envy. Using (18) with Proposition 3.2 and the proof thereof, one obtains

$$V_r(\pi) \geqslant \frac{n - r + 1}{n - r + 2} \frac{e(\pi^{**})^2}{n}, r \in \mathcal{R},$$

Electronic copy available at: https://ssrn.com/abstract=3512113

and

$$V(\pi) = \frac{\sum_{k=1}^{n} V_k(\pi)}{n} \geqslant \frac{\sum_{k=1}^{r} V_k(\pi)}{n} \geqslant r\left(\frac{n-r+1}{n-r+2}\right)\frac{e(\pi^{**})^2}{n^2}$$
$$\geqslant \min_{2\leqslant r\leqslant n} r\left(\frac{n-r+1}{n-r+2}\right)\frac{e(\pi^{**})^2}{n^2} = 2\left(\frac{n-1}{n}\right)\frac{e(\pi^{**})^2}{n^2},$$

from which (17) follows.

Notice that the upper bound in (17) is attained in the following case. Consider a single item to be allocated among n agents.

The value placed on the item by the first two agents is M, while all other agents give a value $\xi > 0$ arbitrarily close to 0. A minimum variance allocation is obtained by providing the only item to any k-th agent. In this case, one the minimal variance can be made arbitrarily close to $V^* = \frac{2(n-1)}{n^3}M^2$, and the maximal envy attained by either of the first two agents, is represented by $e(\pi^{**}) = M = n\sqrt{\frac{n}{2(n-1)}V^*}$.

References

Bouveret, S., Y. Chevaleyre, N. Maudet, and H. Moulin (2016). Fair allocation of indivisible goods. In F. Brandt, V. Conitzer, U. Endriss, J. Lang, and A. D. Procaccia (Eds.), *Handbook of Computational Social Choice*, pp. 284–310. Cambridge University Press.

Bouveret, S. and M. Lemaître (2016). Characterizing conflicts in fair division of indivisible goods using a scale of criteria. *Autonomous Agents and Multi-Agent Systems 30*(2), 259–290.

Bouveret, S., M. Lemaître, H. Fargier, and J. Lang (2005). Allocation of indivisible goods: a general model and some complexity results. In *Proceedings of the 4th international joint conference on Autonomous Agents and Multiagent Systems*, pp. 1309–1310.

Brams, S. J., M. A. Jones, and C. Klamler (2013). N-person cake-cutting: There may be no perfect division. *The American Mathematical Monthly 120*(1), 35–47.

Brams, S. J. and A. D. Taylor (1996). *Fair Division: From Cake-Cutting to Dispute Resolution*. Cambridge University Press.

Caragiannis, I., D. Kurokawa, H. Moulin, A. D. Procaccia, N. Shah, and J. Wang (2019). The unreasonable fairness of maximum Nash welfare. *ACM Transactions on Economics and Computation (TEAC) 7* (3), 1–32.

Chevaleyre, Y., U. Endriss, and N. Maudet (2017). Distributed fair allocation of indivisible goods. *Artificial Intelligence 242*, 1–22.

Foley, D. (1967). Resource allocation and the public sector. *Yale Econ. Essays 7*, 45–98.

Ghodsi, M., M. Hajiaghayi, M. Seddighin, S. Seddighin, and H. Yami (2018). Fair allocation of indivisible goods: Improvements and generalizations. In *Proceedings of the 2018 ACM Conference on Economics and Computation*, pp. 539–556.

Electronic copy available at: https://ssrn.com/abstract=3512113

Hardy, G. H., J. E. Littlewood, and G. Pólya (1952). *Inequalities*. Cambridge, at the University Press. 2d ed.

Kilgour, D. M. and R. Vetschera (2018). Two-player fair division of indivisible items: Comparison of algorithms. *European Journal of Operational Research 271*(2), 620–631.

Kurokawa, D., A. D. Procaccia, and J. Wang (2018). Fair enough: Guaranteeing approximate maximin shares. *J. ACM 65*(2), Article 8, 27 pages.

Lang, J. and J. Rothe (2016). Fair division of indivisible goods. In J. Rothe (Ed.), *Economics and Computation*, pp. 493–550. Springer-Verlag, Berlin.

Lipton, R., E. Markakis, E. Mossel, and A. Saberi (2004). On approximately fair allocations of indivisible goods. In *Proceedings of the 5th ACM conference on Electronic Commerce*, pp. 125–131.

Moulin, H. (2019). Fair division in the internet age. *Annual Review of Economics 11*, 407–441.

Procaccia, A. D. (2016). Cake cutting algorithms. In F. Brandt, V. Conitzer, U. Endriss, J. Lang, and A. D. Procaccia (Eds.), *Handbook of Computational Social Choice*, pp. 311–330. Cambridge University Press.

Robertson, J. and W. Webb (1998). *Cake-Cutting Algorithms: Be Fair If You Can*. A.K. Peters, Natick, MA.

Thomson, W. (2011). Fair allocation rules. In K. J. Arrow, A. Sen, and K. Suzumura (Eds.), *Handbook of Social Choice and Welfare*, Volume 2, pp. 393 – 506. North Holland, Amsterdam.

Vetschera, R. (2010). A general branch-and-bound algorithm for fair division problems. *Computers & Operations Research 37*(12), 2121–2130.

Electronic copy available at: https://ssrn.com/abstract=3512113