



**HAL**  
open science

## Generalized binary utility functions and fair allocations

Franklin Camacho, Rigoberto Fonseca-Delgado, Ramón Pino Pérez, Guido Tapia

► **To cite this version:**

Franklin Camacho, Rigoberto Fonseca-Delgado, Ramón Pino Pérez, Guido Tapia. Generalized binary utility functions and fair allocations. *Mathematical Social Sciences*, 2023, 121, pp.50-60. 10.1016/J.MATHSOCSCI.2022.10.003 . hal-04452972

**HAL Id: hal-04452972**

**<https://hal.science/hal-04452972>**

Submitted on 12 Feb 2024

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Generalized binary utility functions and fair allocations

Franklin Camacho<sup>a,\*</sup>, Rigoberto Fonseca-Delgado<sup>a</sup>, Ramón Pino Pérez<sup>b</sup>, Guido Tapia<sup>a</sup>

<sup>a</sup>*School of Mathematical and Computational Sciences, Yachay Tech University, Urcuquí, Ecuador*

<sup>b</sup>*Centre de Recherche en Informatique de Lens (CRIL), CNRS and Université d'Artois, Lens, France*

---

## Abstract

The problem of finding envy-free allocations of indivisible goods cannot always be solved; therefore, it is common to study some relaxations such as envy-free up to one good (EF1) and envy-free up to any positively valued good (EFX). Another property of interest for the efficiency of an allocation is the Pareto Optimality (PO). Under additive utility functions for goods, it is possible to find EF1 and PO allocations using the Nash social welfare. However, finding an allocation that maximizes the Nash social welfare is a computationally costly problem. Maximizing the utilitarian social welfare subject to EF1 constraints is an NP-complete problem for the case where three or more agents participate. In this work, we propose a restricted case of additive utility functions called generalized binary utility functions. The proposed utilities are a generalization of binary and identical utilities simultaneously. In this scenario, we present a polynomial-time algorithm that maximizes the utilitarian social welfare and, at the same time, produces an EF1 and PO allocation for goods as well as for chores. Moreover, a slight modification of our algorithm gives a better allocation: one which is EFX.

*Keywords:* Allocation of indivisible goods/chores, envy-free up to one good/chore, efficiency, additive utility function.

---

## 1. Introduction

The resource allocation problem has been widely studied in mathematics and economics for almost a century, [17, 18, 15, 3, 9, 8]. The main elements

---

\*Corresponding author.

*Email addresses:* [fcamacho@yachaytech.edu.ec](mailto:fcamacho@yachaytech.edu.ec) (Franklin Camacho), [rfonseca@yachaytech.edu.ec](mailto:rfonseca@yachaytech.edu.ec) (Rigoberto Fonseca-Delgado), [pinoperez@cril.fr](mailto:pinoperez@cril.fr) (Ramón Pino Pérez), [guido.tapia@yachaytech.edu.ec](mailto:guido.tapia@yachaytech.edu.ec) (Guido Tapia)

in the problem are agents and resources. The goal is to distribute (or allocate) the resources among the agents in a “good manner.” The agents can represent individuals, objects, government institutions, among others, depending on the application. The set of resources or goods to be distributed can be divisible or indivisible. In general, these sets are considered finite. In this work, we study the problem of allocating indivisible resources among a group of agents, with the aim of satisfying both the group and each individual in the best possible way.

This topic has many applications, for instance in solving divorce disputes, dividing an inheritance, sharing apartment rents, or even assigning household chores. In the last decade, there has been a considerable interest in the computational aspects of this problem. In particular, in Artificial Intelligence and more specifically in MultiAgent Systems, these problems are studied and renamed MultiAgent Resource Allocations (MARA) problems [2, 10].

Finding a correct distribution of resources consists of distributing all the resources among the agents fairly and efficiently. To establish efficiency and some criterion of fairness, it is necessary to consider the preferences that each agent has over resources. In general, these preferences over resources are established through additive utility functions.

Traditionally, fairness is established through properties such as envy-freeness or proportionality. However, there are situations where it is impossible to find allocations that meet any of these properties. Thus, other weaker versions of fairness, such as envy-free up to one good [7] or proportionality up to one good [11] are considered. Although there are results that, under certain conditions, guarantee the existence of allocations with some fairness property (see [9]), finding them is a computationally complicated problem [12]. Just considering fairness may not be enough, because it could imply loss in group satisfaction.

Efficiency, also known as Pareto efficiency or Pareto optimality, is related to the group satisfaction by an allocation. One way to find efficient allocations is through social welfare functions. Caragianis et al, [9], showed that, under additive utility functions for goods, it is always possible to find an allocation that is Pareto optimal and envy-free up to one good. Actually, they prove that some of the allocations that maximize the Nash social welfare are Pareto optimal and envy-free up to one good. Unfortunately, finding allocations which maximize the Nash welfare is also an NP-hard problem [16] (see also [2]).

Searching allocations which maximize the utilitarian social welfare is in general a more tractable problem from the computational point of view and, because of that, commonly used. A well-known result is that, under additive utility functions, allocations that maximize this social welfare are Pareto optimal (see Theorem 1), although the converse is not true. Moreover, allocations that maximize this social welfare do not always satisfy fairness properties. In Example 2, an allocation that is Pareto optimal and does not maximize utilitarian social welfare is proposed; besides, we find, in this example, that no allocation that maximizes utilitarian social welfare is envy free up to one good.

When we consider additive utility functions, it is possible to define a procedure that allows to find all the allocations which maximize utilitarian social wel-

fare, see [8]. Moreover, finding these allocations is a computationally tractable problem. Actually, in this work we propose a very simple algorithm in polynomial time, for scenarios of additive utilities, that finds an allocation which maximizes the utilitarian social welfare for goods as well as for chores.

Unlike the allocations that maximize the Nash social welfare which are EF1, the allocations that maximize the utilitarian social welfare are not, in general, EF1. Moreover, there are additive scenarios in which the property EF1 fails for every allocation which maximizes the utilitarian social welfare (see Example 2). Indeed, in this example we can see that Nash and utilitarian social welfares are independent.

Finding an allocation that maximizes the utilitarian social welfare and satisfies the EF1 criteria is an NP-complete problem when the number of agents is greater than or equal to three (see [4]). However, for some specialized scenarios, there are very simple algorithms in polynomial time which find EF1, and even EFX allocations. That is the case of identical utilities (see Barman et al. [5]) and others like bivaluated utilities (Ebadian et al. [14]) or binary utilities [5] (see Section 6).

In this work, we consider a class of additive functions, called generalized binary utility functions. This class is more general than the classes of binary and identical utilities. Intuitively, each resource has a market-price; each agent either does not want the resource at all, or wants it and values it by its market price. In the framework of these utility functions, the following results are established:

1. A characterization for Pareto optimal allocations (Theorem 3).
2. Each allocation that maximizes the Nash social welfare also maximizes the utilitarian social welfare (Theorem 4).
3. Constructive proofs of the existence of allocations which maximize the utilitarian social welfare, which are PO and respectively EF1 and EFX for goods and for chores. These allocations are obtained in polynomial time (Theorems 7 and 8).

Moreover, we propose a basic algorithm in  $O(nm)$  which finds, under additive utility functions, an allocation that maximizes the utilitarian social welfare (Theorem 6).

The rest of this work is organized as follows. Section 2 introduces the concepts and problems studied in this paper. Section 3 is devoted to a characterization of Pareto Optimality in our particular scenarios. In Section 4, we propose a very simple and tractable algorithm for maximizing the utilitarian social welfare and we study its justice properties in the case of generalized binary utilities. In Section 5, we give a slight generalization of our generalized binary scenarios and prove that most of the results obtained in the previous sections don't hold for this new class of scenarios. Section 6 contains a comparison of our work with other works using specialized scenarios for tackling the problems of fairness and efficiency. We conclude in Section 7 with some final remarks. The proofs, our detailed algorithm and some detailed examples can be found in Appendix A.

## 2. Preliminaries

The set of agents is denoted by  $\mathcal{A} = \{1, \dots, n\}$  and the set of resources is denoted by  $\mathcal{R} = \{r_1, \dots, r_m\}$ . So,  $|\mathcal{R}| = m$  and  $|\mathcal{A}| = n$ . An allocation of resources is a function  $F : \mathcal{R} \rightarrow \mathcal{A}$ . For each agent  $i$ ,  $F^{-1}(i) = \{r \in \mathcal{R} : F(r) = i\}$  is the set of resources (or bundle) assigned to  $i$ . The set of all possible allocations is denoted by  $\mathcal{A}^{\mathcal{R}}$ . The number of possible allocations depends on  $n$  and  $m$ , given that  $|\mathcal{A}^{\mathcal{R}}| = |\mathcal{A}|^{|\mathcal{R}|} = n^m$ . The set of all subsets of  $\mathcal{R}$  is denoted by  $\mathcal{P}(\mathcal{R})$ . We consider two kinds of resources: *goods* and *chores*. Goods are resources that agents are supposed to accept and chores are resources that agents are supposed to reject. We will assume that either all resources are goods or all resources are chores.

The preference of agents over resources is established through utility functions which are additive, that is, functions of the type  $u : \mathcal{P}(\mathcal{R}) \rightarrow \mathbb{R}$  which satisfy:

- $u(\emptyset) = 0$ ;
- $\forall S \in \mathcal{P}(\mathcal{R})$  with  $S \neq \emptyset$ ,  $u(S) = \sum_{s \in S} u(\{s\})$ .

When the resources are goods, we always have  $\forall S \in \mathcal{P}(\mathcal{R})$ ,  $u(S) \geq 0$ . When the resources are chores, we always have  $\forall S \in \mathcal{P}(\mathcal{R})$ ,  $u(S) \leq 0$ . For each  $i \in \mathcal{A}$ ,  $u_i$  denotes the additive utility function associated to  $i$ . For simplicity,  $u(\{s\})$  will be denoted by  $u(s)$ . If every agent establishes an additive utility function in a problem of indivisible resource allocation, we say that it is an additive scenario.

**Definition 1.** *Assume that for each  $r_k \in \mathcal{R}$  we have a mapping  $r_k \mapsto p_k$  where  $p_k$  is a real number different from 0. Let  $u$  be a function  $u : \mathcal{P}(\mathcal{R}) \rightarrow \mathbb{R}$ . We say that  $u$  is a generalized binary (g-binary for short) utility function if:*

- $u$  is an additive utility function;
- for every  $r_k \in \mathcal{R}$ ,  $u(r_k) \in \{0, p_k\}$ .

We say that we are in a goods (chores) g-binary scenario when all the resources are goods (chores) and every agent of the resource allocation problem has a utility function which is g-binary for the same mapping  $r_k \mapsto p_k$ .

Note that in case of goods (chores)  $p_k$  can be viewed as the market-price (rejection intensity) of the resource  $r_k$ . The value 0 does not change an agent's utility; in case of goods, this value can be interpreted as a reject, whereas in chores, it can be interpreted as a preference.

Clearly, if  $u$  is an additive utility function such that for each  $r_k \in \mathcal{R}$ ,  $u(r_k) \in \{0, 1\}$ , then  $u$  is a g-binary utility function. This type of function is known as binary utility function, see [1]. Thus, the goods g-binary scenarios are a generalization of binary scenarios.

We assume that in goods and chores scenarios there is not a resource  $r$  such that for every agent  $i \in \mathcal{A}$ ,  $u_i(r) = 0$ . Therefore, in a goods g-binary scenario,  $\forall r_j \in \mathcal{R}$ ,  $\max\{u_i(r_j) : i \in \mathcal{A}\} = p_{r_j} > 0$  and  $\min\{u_i(r) : i \in \mathcal{A}\} = 0$ .

While, in a chores g-binary scenario,  $\forall r_j \in R, \max\{u_i(r_j) : i \in A\} = 0$  and  $\min\{u_i(r) : i \in A\} = p_{r_j} < 0$ .

A goods g-binary scenario in which the value 0 is not taken by the utility functions is called a scenario of identical utility functions. These scenarios were considered by Barman et al. [5]. There are examples where imposing a scenario of identical utilities is not very adequate to find an allocation that best satisfies the agents. The following example shows such a situation.

**Example 1.** *Suppose there are two goods  $r_1, r_2$  and two agents 1, 2. Agent 1 wants  $r_1$  and his utility for this good is  $p_1$ , that is  $u_1(r_1) = p_1$ . Agent 1 does not want  $r_2$  and his utility for this good is null, that is  $u_1(r_2) = 0$ . For agent 2, it is all the contrary, more precisely: 2 wants  $r_2$  and his utility for this good is  $p_2$ , that is  $u_2(r_2) = p_2$  but 2 does not want  $r_1$  and his utility for this good is null, that is  $u_2(r_1) = 0$ . In such a g-binary scenario, it is clear that the best allocation corresponds to allocating  $r_1$  to 1 and  $r_2$  to 2. In a scenario with identical utilities the agents can't express that they are not interested in a good (utility 0). Thus the allocation which gives  $r_1$  to 2 and  $r_2$  to 1, in a scenario of identical utilities produces the same social utilitarian welfare but this allocation is far from satisfying the agents.*

Note that the goods g-binary scenarios are a simplification of the Fisher market model in economy [6].

### 2.1. Fairness, efficiency, and social welfare

An attractive fairness criterion in additive scenarios is the absence of envy. If no agent strictly prefers the bundle assigned to another agent instead of its own bundle, the allocation is envy-free. More precisely, an allocation  $F$  is **envy-free** (EF) if  $\forall i, j \in \mathcal{A}, u_i(F^{-1}(i)) \geq u_i(F^{-1}(j))$ . If there exists an agent  $i \in \mathcal{A}$  such that  $u_i(F^{-1}(i)) < u_i(F^{-1}(j))$  for some  $j \in \mathcal{A}$ , then the agent  $i$  envies the agent  $j$ . The property EF is the most desirable property, but in a simple example of an indivisible resource with two agents it is impossible to find an allocation without envy. In the literature [7, 9, 1, 13, 14], weaker versions of the envy free property can be found. The weakest among these is envy free up to one good. The following definition establishes the main weak envy free notions for goods and chores.<sup>1</sup>

**Definition 2.** *Let  $F$  be in  $\mathcal{A}^{\mathcal{R}}$ ,*

1.  $F$  is an **envy-free up to one resource** (EF1) allocation for goods if  $\forall i, j \in \mathcal{A}, \exists g \in F^{-1}(j)$  such that

$$u_i(F^{-1}(i)) \geq u_i(F^{-1}(j) \setminus \{g\}) \quad (1)$$

---

<sup>1</sup>Actually, one can give a unified and more compact definition but it is clearer if we split it into two cases: goods and chores.

2.  $F$  is an **envy-free up to one resource (EF1) allocation for chores** if  $\forall i, j \in \mathcal{A}, \exists g \in F^{-1}(i)$  such that

$$u_i(F^{-1}(i) \setminus \{g\}) \geq u_i(F^{-1}(j)) \quad (2)$$

3.  $F$  is an **envy-free up to any non zero valued resource (EFX) allocation for goods** if  $\forall i, j \in \mathcal{A}$ ,

$$\forall g \in F^{-1}(j) \text{ with } u_i(g) > 0, \quad u_i(F^{-1}(i)) \geq u_i(F^{-1}(j) \setminus \{g\}) \quad (3)$$

4.  $F$  is an **envy-free up to any non zero valued resource (EFX) allocation for chores** if  $\forall i, j \in \mathcal{A}$ ,

$$\forall g \in F^{-1}(i) \text{ with } u_i(g) < 0, \quad u_i(F^{-1}(i) \setminus \{g\}) \geq u_i(F^{-1}(j)) \quad (4)$$

5.  $F$  is an **envy-free up to any valued resource (EFX<sub>0</sub>) allocation for goods** if  $\forall i, j \in \mathcal{A}$ ,

$$\forall g \in F^{-1}(j), \quad u_i(F^{-1}(i)) \geq u_i(F^{-1}(j) \setminus \{g\}) \quad (5)$$

6.  $F$  is an **envy-free up to any valued resource (EFX<sub>0</sub>) allocation for chores** if  $\forall i, j \in \mathcal{A}$ ,

$$\forall g \in F^{-1}(i), \quad u_i(F^{-1}(i) \setminus \{g\}) \geq u_i(F^{-1}(j)) \quad (6)$$

It is clear that in the case of additive utilities for goods or for chores we have:

$$EF \Rightarrow EFX_0 \Rightarrow EFX \Rightarrow EF1$$

The efficiency, also known as Pareto efficiency or Pareto optimality, aims at characterizing when the allocation best satisfies the group. Remember that if  $F$  and  $G$  are allocations in  $\mathcal{A}^{\mathcal{R}}$ , we say that  $F$  is *Pareto dominated* by  $G$ , when:

- $\forall i \in \mathcal{A}, u_i(F^{-1}(i)) \leq u_i(G^{-1}(i))$  and
- $\exists j \in \mathcal{A}$  such that  $u_j(F^{-1}(j)) < u_j(G^{-1}(j))$

We say that  $G$  is *Pareto optimal* (PO), if it is not Pareto dominated by another allocation.

One way to measure the social satisfaction of the agents is through the Nash and utilitarian social welfare functions. Let us recall their precise definitions. The utilitarian social welfare of  $F$ , denoted by  $SW_u(F)$ , is defined by

$$SW_u(F) = \sum_{i \in \mathcal{A}} u_i(F^{-1}(i)) \quad (7)$$

we put  $MSW_u = \{F : SW_u(F) \geq SW_u(G), \forall G \in \mathcal{A}^{\mathcal{R}}\}$ . The Nash social welfare, denoted by  $SW_{Nash}(F)$ , is defined by

$$SW_{Nash}(F) = \prod_{i \in \mathcal{A}} u_i(F^{-1}(i)) \quad (8)$$

we put  $MSW_{Nash} = \{F : SW_{Nash}(F) \geq SW_{Nash}(G), \forall G \in \mathcal{A}^{\mathcal{R}}\}$ .

A well-known result is that any allocation that maximizes utilitarian social welfare is PO:

**Theorem 1.** *Under an additive scenario, let  $F$  be in  $\mathcal{A}^{\mathcal{R}}$ , if  $F \in MSW_u$ , then  $F$  is PO.*

Although in general it is not always possible to find EF allocations, fortunately, Caragiannis and colleagues [9] showed that under an additive scenario for goods it is possible to find EF1 and PO resource allocations. Actually, they prove the following theorem:

**Theorem 2.** *[Caragiannis et al. [9]] Under an additive scenario for goods, every allocation<sup>2</sup> that maximizes Nash social welfare is EF1 and PO.*

Note that, with the help of Theorem 2, it is easy to see that the converse of Theorem 1 is false. The following example shows this.

**Example 2.** *Let  $\mathcal{R} = \{r_1, r_2\}$  be the set of resources and  $\mathcal{A} = \{1, 2\}$  the set of agents where each agent  $i$  establishes its utility function  $u_i$  over every resource, according to Table 1.*

Table 1: Utility functions.

	$r_1$	$r_2$
$u_1$	10	10
$u_2$	3	2

*Let  $F$  and  $G$  be the allocations defined by Table 2.*

Table 2: Allocations.

	$r_1$	$r_2$
$F$	1	1
$G$	2	1

*The utility by bundle received by each agent and social welfare are showed in Table 3.*

Table 3: Utility by bundle received and social welfare.

	1	2	$SW_u$	$SW_{Nash}$
$u_i(F^{-1}(i))$	20	0	20	0
$u_i(G^{-1}(i))$	10	3	13	30

<sup>2</sup>Actually, the Theorem as stated is true when the maximum Nash social welfare is strictly positive. When it is zero, it is necessary to impose that the set of agents having a good is a maximal set (see [9]).



It is easy to see that  $F$  is the only allocation which maximizes utilitarian social welfare. However,  $F$  is not EF1 because agent 2 envies agent 1,  $u_2(F^{-1}(2)) = 0 < 5 = u_2(F^{-1}(1))$ , but even after removing  $r_1$  or  $r_2$  from  $(F^{-1}(1))$ , envy does not disappear:

$$\begin{aligned} u_2(F^{-1}(2)) &= 0 < 2 = u_2(F^{-1}(1) \setminus \{r_1\}) \text{ and} \\ u_2(F^{-1}(2)) &= 0 < 3 = u_2(F^{-1}(1) \setminus \{r_2\}) \end{aligned}$$

The maximum Nash social welfare is reached at 30, so  $G$  maximizes the Nash social welfare. By Theorem 2  $G$  is EF1, but it does not maximize the utilitarian social welfare. In the following table, we identify the properties that satisfy  $F$  and  $G$ ; if an allocation satisfies a property we use  $\checkmark$  and  $\times$  otherwise.

Table 4: Identification of properties (within a non g-binary scenario).

	PO	EF1	$MSW_u$	$MSW_{Nash}$
$F$	$\checkmark$	$\times$	$\checkmark$	$\times$
$G$	$\checkmark$	$\checkmark$	$\times$	$\checkmark$

### 3. A characterization of Pareto Optimality

The following theorem shows that under g-binary scenarios, the converse of Theorem 1 is satisfied; i.e., having the property PO and belonging to  $MSW_u$  are equivalent.

**Theorem 3.** *Assume a goods (chores) g-binary scenario and let  $F$  be an allocation in  $\mathcal{A}^R$ . Then,  $F$  is PO if, and only if,  $F \in MSW_u$ .*

Note that, in general additive scenarios, Theorem 3 does not hold as Example 2 reveals.

We have seen in Example 2 that  $MSW_{Nash} \not\subseteq MSW_u$ . However, under goods g-binary scenarios, Theorems 2 and 3 together, tell us that  $MSW_{Nash} \subseteq MSW_u$ . This is important and will be stated in the following result.

**Theorem 4.** *Under a goods g-binary scenario, each allocation that maximizes Nash social welfare also maximizes utilitarian social welfare.*

A straightforward corollary of Theorem 4 and Theorem 2, is the existence, under goods g-binary scenarios, of allocations maximizing the utilitarian social welfare and satisfying the EF1 property. The existence is based in finding an allocation producing a maximum Nash welfare, a hard problem from a computational point of view. In the next section we will see that in g-binary scenarios it is easy to compute allocations EF1 producing a maximal utilitarian welfare.

### 4. A simple Algorithm and its behavior in some additive scenarios

In this section, we propose a very natural and simple algorithm and we analyze its behavior in additive scenarios, in particular in g-binary scenarios.

Let us start with a key result in the conception of the algorithm. It establishes that an allocation distributes resources to the agents that maximize them if, and only if, this allocation maximizes the utilitarian social welfare. One can find a straightforward proof of this result<sup>3</sup>, using a matrix approach, in [8].

**Theorem 5.** *Under an additive scenario,  $F \in MSW_u$  if, and only if,  $\forall i \in \mathcal{A}, \forall r \in F^{-1}(i)$  we have that  $u_i(r) = \max\{u_j(r) : j \in \mathcal{A}\}$ .*

Suppose that in an additive scenario,  $\mathcal{R} = \{r_1, \dots, r_m\}$  (all the resources are goods or all the resources are chores) and  $\mathcal{A} = \{1, \dots, n\}$ . Let  $\alpha_1, \dots, \alpha_m$  be the real numbers defined in the following way: for each  $r_k \in \mathcal{R}$ ,  $\alpha_k = \max\{u_j(r_k) : j \in \mathcal{A}\}$ . Then, by the previous theorem the following simple algorithm defines an allocation,  $\Gamma$ , having a maximal utilitarian welfare, that is,  $\Gamma \in MSW_u$ .

---

**Algorithm 1:** An allocation for maximal utilitarian welfare

---

**Input:** Two finite sets,  $\mathcal{R} = \{r_1, \dots, r_m\}$  for the resources,

$\mathcal{A} = \{1, \dots, n\}$  for the agents, and their respective utilities  $u_i$

**Output:** The allocation  $\Gamma$

1  $v_0 \leftarrow (0, \dots, 0)$

2 **for**  $k \leftarrow 1$  **to**  $m$  **do**

3      $\alpha_k \leftarrow \max\{u_j(r_k) : j \in \mathcal{A}\}$

4      $P_k \leftarrow \{j \in \mathcal{A} : u_j(r_k) = \alpha_k\}$

5      $l_k \leftarrow \min\{|[v_{k-1}]_j| : j \in P_k\}$

6      $M_k \leftarrow \{i \in P_k : |[v_{k-1}]_i| = l_k\}$

7      $j_k \leftarrow \min\{M_k\}$

8      $\Gamma(r_k) \leftarrow j_k$

9      $[v_k]_i \leftarrow \begin{cases} [v_{k-1}]_i + u_{j_k}(r_k), & \text{if } i = j_k \\ [v_{k-1}]_i, & \text{if } i \neq j_k. \end{cases}$  for all  $i \in \mathcal{A}$

10 **return**  $\Gamma$

---

The idea of Algorithm 1 is very simple: the resource  $r_k$  is allocated to an agent who maximizes its utility and such that before this step, that is until the partial allocation of resources  $\{r_1, \dots, r_{k-1}\}$  is done, he has the minimal utility in case of goods and the maximal utility in the case of chores. A detailed description of the behavior of this algorithm appears in the Appendix.

From Theorem 1 we have the following result:

**Corollary 1.** *Under an additive scenario,  $\Gamma$  is PO.*

We saw in Example 2 that the allocation  $F$ , the unique allocation that maximizes the utilitarian social welfare is not EF1 (see Table 4). But this allocation is indeed the allocation  $\Gamma$ , thus in general, Algorithm 1 does not produce an allocation EF1.

---

<sup>3</sup>Actually, in [8] the *if* is proved. The *only if* is obvious.

The following observation summarizes some facts in g-binary scenarios. This will be proved throughout Example 5 in Appendix A.

**Remark 1.** *It is important to note that in g-binary scenarios there are allocations EF1 which are not in  $MSW_u$ . In those scenarios there are allocations which are PO and  $MSW_u$  but they are neither EF1 nor belong to  $MSW_{Nash}$ . Also in those scenarios  $\Gamma$  is EF1, PO, it is in  $MSW_u$  but, in general, it is not in  $MSW_{Nash}$ . It can happen that there exist allocations that are EF1, PO, and they are in both  $MSW_u$  and in  $MSW_{Nash}$ .*

We have already said that Algorithm 1 produces a maximal utilitarian welfare. This occurs in case of additive scenarios and due to Theorem 5 and the definition of Algorithm 1. The next result summarizes this and gives the complexity of the algorithm.

**Theorem 6.** *Under a goods (chores) additive scenario,  $\Gamma \in MSW_u$  and it is obtained in  $O(nm)$  operations.*

In addition to this, we will show next that in g-binary scenarios for goods and chores, the allocation  $\Gamma$ , given by Algorithm 1 is also envy-free up to one resource (good or chore).

**Theorem 7.** *Under a g-binary scenario for goods or chores, the allocation  $\Gamma$  given by Algorithm 1 produces a maximal utilitarian welfare, is EF1 and PO. Its run time is  $O(mn)$ .*

As a matter of fact, in a g-binary scenario for chores, if for any resource  $r$ ,

$$\max_{i \in \mathcal{A}} \{u_i(r)\} \neq \min_{i \in \mathcal{A}} \{u_i(r)\}$$

we have that  $\Gamma$  is EF. That is because, in such a case,  $\Gamma$  is an allocation in which all the agents give utility 0 to their bundles.

Moreover, with a very slight modification of Algorithm 1, we will obtain another allocation,  $\Gamma^*$ , which is EFX. This modification consists in taking one more step: reordering the resources. In the case of goods we order the resources in decreasing order and in the case of chores we order the resources in increasing order according to the maximal absolute value of utilities given to resources. More precisely, a resource  $r$  precedes another resource  $r'$  if  $\max_{i \in \mathcal{A}} \{|u_i(r)|\} \geq \max_{i \in \mathcal{A}} \{|u_i(r')|\}$ . For this  $\Gamma^*$  we have the following result.

**Theorem 8.** *Under a g-binary scenario for goods or chores, the allocation  $\Gamma^*$  given by Algorithm 1 modified as previously indicated maximizes the utilitarian social welfare, is EFX and PO. Moreover, its run time is  $O(m \log m + mn)$ .*

It is interesting to observe that the algorithm producing  $\Gamma^*$  is quite similar to Barman et al. algorithm [5] producing an allocation EFX, in the case of identical scenarios for goods. In fact, in identical scenarios, the notions EFX and  $EFX_0$  coincide. This is not the case in g-binary scenarios. This can be viewed by building a g-binary scenario, in which  $\Gamma^*$ , the allocation given by the

modified Algorithm 1, which is EFX, is not EFX<sub>0</sub> (see Example 6 in Appendix A).

Note that if  $F$  is an allocation EFX for goods (for chores) such that for all  $i, j \in \mathcal{A}$  such that  $u_i(F^{-1}(i)) < u_i(F^{-1}(j))$  we have that  $F^{-1}(j) \subseteq \{r \in \mathcal{R} : u_i(r) > 0\}$  ( $F^{-1}(i) \subseteq \{r \in \mathcal{R} : u_i(r) < 0\}$ ), then  $F$  is EFX<sub>0</sub> for goods (for chores).

## 5. Some limits of Algorithm 1

It is natural to ask if Algorithm 1 has other interesting properties. For instance, does it compute a maximal egalitarian welfare<sup>4</sup> allocation? We will see in Example 3 that it is not the case even when we work in g-binary scenarios.

Another question concerns the properties of Algorithm 1 in a slight modification of g-binary scenarios. Remember that in g-binary scenarios, every Pareto efficient allocation indeed maximizes utilitarian social welfare (see Theorem 3), and as a consequence, in the case of goods, every assignment that maximizes Nash's social welfare also maximizes the utilitarian one (see Theorem 5). Thus, natural questions are: in these modified scenarios, is it possible to preserve these properties? Furthermore, does Algorithm 1 achieve fair allocations?

The following example shows that neither  $\Gamma$  nor  $\Gamma^*$  maximize the egalitarian social welfare.

**Example 3.** *There are three agents and three resources. The utilities are given in Table 5. In Table 6 are the allocations. In Table 7 appear the utilities of each agent for every allocation and the egalitarian social welfare denoted by  $SW_e$ .*

Table 5: Utility functions.

	$r_1$	$r_2$	$r_3$
$u_1$	4	1	2
$u_2$	4	0	2
$u_3$	4	0	0

Table 6: Allocations.

	$r_1$	$r_2$	$r_3$
$\Gamma$	1	1	2
$\Gamma^*$	1	1	2
$A$	3	1	2

<sup>4</sup>The egalitarian social welfare of an allocation  $F$ , denoted by  $SW_e(F)$  is defined by  $SW_e(F) = \min\{u_i(F^{-1}(i)) : i \in \mathcal{A}\}$ .

Table 7: Utility by bundle received and egalitarian social welfare.

	1	2	3	$SW_e$
$u_i(\Gamma^{-1}(i))$	5	2	0	0
$u_i(\Gamma^{*-1}(i))$	5	2	0	0
$u_i(A^{-1}(i))$	1	2	4	1

Table 7 shows that  $\Gamma$  and  $\Gamma^*$  do not maximize the egalitarian social welfare because the allocation  $A$  has egalitarian social welfare which is better than the egalitarian social welfare of  $\Gamma$  and  $\Gamma^*$ .

Now, we propose a scenario called  $\epsilon$ -g-binary, which slightly changes the range of the utilities of each agent, generalizing the notion of g-binary scenarios. Let us see the definition below.

**Definition 3.** Assume that for each  $r_k \in \mathcal{R}$  we have a mapping  $r_k \mapsto p_k$  where  $p_k$  is a real number. Let  $u$  be a function  $u : \mathcal{P}(\mathcal{R}) \rightarrow \mathbb{R}$ . We say that  $u$  is an  $\epsilon$ -g-binary utility function for goods (chores) if:

- $\epsilon \geq 0$  ( $\epsilon \leq 0$ );
- $u$  is an additive utility function;
- for every  $r_k \in \mathcal{R}$ ,  $p_k > \epsilon$  ( $p_k < \epsilon$ ) and  $u(r_k) \in \{\epsilon, p_k\}$ .

When every agent has an  $\epsilon$ -g-binary utility function for goods (chores) with the same mapping  $r_k \mapsto p_k$ , we say that the scenario is goods (chores)  $\epsilon$ -g-binary.

The above definition generalizes Definition 1; in the case goods (chores),  $\epsilon$  is the minimum (maximum) value for the scenario. It means that the minimum (maximum) value could be equal to zero or greater than (less than) zero. When  $\epsilon$  is equal to 0, we get exactly the goods (chores) g-binary scenario.

In the following example, we give an  $\epsilon$ -g-binary scenario, with  $\epsilon > 0$ , for which there exists an allocation  $G \notin MSW_u$  having a greater Nash social welfare than all the allocations that maximize the utilitarian social welfare.

**Example 4.** We consider  $n = m = 3$ . Table 8 defines our  $\epsilon$ -g-binary scenario with  $b > a > \epsilon > 0$ .

Table 8: Utility functions.

	$r_1$	$r_2$	$r_3$
$u_1$	$b$	$a$	$a$
$u_2$	$b$	$\epsilon$	$\epsilon$
$u_3$	$b$	$\epsilon$	$\epsilon$

Note that any allocation that maximizes the utilitarian social welfare assigns the resources  $r_2$  and  $r_3$  to agent 1 (see Theorem 5). Thus, there are only three allocations,  $F_1, F_2$  and  $F_3$ , which maximize the utilitarian social welfare. They

are shown in table 9. This table shows also another allocation, namely the allocation  $G$  that assigns the resource  $r_3$  to agent 3, who gives it a utility of  $\epsilon$ .

Table 9: Definition of allocations.

	$r_1$	$r_2$	$r_3$
$F_1$	1	1	1
$F_2$	2	1	1
$F_3$	3	1	1
$G$	2	1	3

Table 10 describes the valuations that the agents give to the received bundle in the different allocations of Table 9; also, it shows the Utilitarian and Nash social welfare of the allocations.

Table 10: Utility by received bundle and social welfare.

	1	2	3	$SW_u$	$SW_{Nash}$
$u_i(F_1^{-1}(i))$	$2a + b$	0	0	$2a + b$	0
$u_i(F_2^{-1}(i))$	$2a$	$b$	0	$2a + b$	0
$u_i(F_3^{-1}(i))$	$2a$	0	$b$	$2a + b$	0
$u_i(G^{-1}(i))$	$a$	$b$	$\epsilon$	$a + b + \epsilon$	$ab\epsilon$

Note that allocation  $G$  has better Nash social welfare than  $F_1, F_2$  and  $F_3$ . This is because the Nash social welfare of  $F_1, F_2$  and  $F_3$  is 0 and the Nash social welfare of  $G$  is strictly positive. Then  $MSW_u \cap MSW_{Nash} = \emptyset$ . Consequently, there are PO allocations that are not contained in  $MSW_u$ ; and therefore, Theorem 3, instantiated in these scenarios, does not hold.

Note that in this example  $\Gamma = \Gamma^* = F_1$ . But, regarding fairness, in allocations  $F_1, F_2$ , and  $F_3$ , at least one agent does not receive any good, and at the same time, this agent values positively the two resources received by another agent. Then, it is not possible to eliminate the envy removing only one resource. Thus, in this scenario, the allocations which maximize utilitarian social welfare do not necessarily satisfy fairness. In particular, Algorithm 1 does not attain success in finding an allocation EF1.

## 6. Related work

Due to the fact that allocations producing a maximal utilitarian welfare are, in additives scenarios, Pareto optimal, in order to assure the Pareto optimality and EF1, it is natural to look for allocations which are in  $MSW_u$  and are simultaneously EF1. However, in general additive scenarios, the problem of finding allocations producing a maximal utilitarian social welfare being simultaneously EF1 is NP-hard [4]. Thus, several works adopt the strategy of working in scenarios in which the utility functions are restricted. Next, we resume the main

proposals studied in the literature, their results and the connections with our framework.

Actually, the work of Caragiannis et al. [9], under general goods additive scenarios shows that some special allocations producing maximum Nash social welfare are EF1 and PO. However, finding this kind of allocations is in general an NP-complete problem. Moreover, this does not solve the problem of finding an allocation that produces a maximal utilitarian social welfare and that is also EF1, because the allocations in  $MSW_{Nash}$  are not necessarily in  $MSW_u$ . However, in g-binary scenarios, we have  $MSW_{Nash} \subseteq MSW_u$ , because in these scenarios being Pareto optimal is equivalent to producing a maximal utilitarian welfare.

In the same vein, Amanatidis et al. [1] connect the maximum of Nash and EFX. They consider  $k$ -valued scenarios, that is, scenarios where there is a set of  $k$  distinct possible values, all real numbers, that agents can assign to their utilities. For 2-valued scenarios they prove that the allocations maximizing the Nash social welfare are  $EFX_0$ . They also prove that for  $k$ -valuated scenarios, with  $k \geq 3$ , it is not true that the allocations maximizing the Nash social welfare are  $EFX_0$ . To the best of our knowledge, there is no known simultaneous existence of EFX and PO allocations under  $k$ -valued scenarios. Note that these scenarios are a generalization of  $\epsilon$ -g-binary scenarios.

Actually, the 2-valued scenarios, called also bivaluated, are deeply studied by Ebadian et al. [14]. They prove that for the case of chores, in that scenario, there exist polynomial time algorithms for calculating allocations which are simultaneously EF1 and PO. They studied also other fairness properties and other scenarios for which they prove the existence of allocations in polynomial time which are fair and PO. Note that the bivaluated scenario and the g-binary are independent.

A particular case of a bivaluated scenario is a binary scenario, where the utility functions can take only the values zero or one. In those scenarios, Barman et al. [5] proposed an algorithm to find an allocation that satisfies EFX and PO constraints and runs in polynomial time. Actually, they propose an algorithm that finds an allocation maximizing the Nash social welfare in polynomial time in the case of binary utilities. In the case of identical utilities, they propose also an algorithm in polynomial time to compute an allocation EFX which maximizes the utilitarian social welfare. They show also that this allocation is a good approximation of a maximum of Nash social welfare.

In Figure 1, we summarize the results of this discussion. The left branch in the figure goes from the most specialized scenario, the binary scenario, to the most general additive scenario passing through the bivaluated and  $k$ -valuated scenarios. The right branch goes from identical scenario, the most specialized in this branch, to additive scenarios (the most general) passing through the new scenarios we study in this work: g-binary and  $\epsilon$ -g-binary scenarios.

We have noted that g-binary scenarios generalize simultaneously binary and identical scenarios and it is easy to see that  $\epsilon$ -g-binary scenarios generalize bivaluated scenarios.

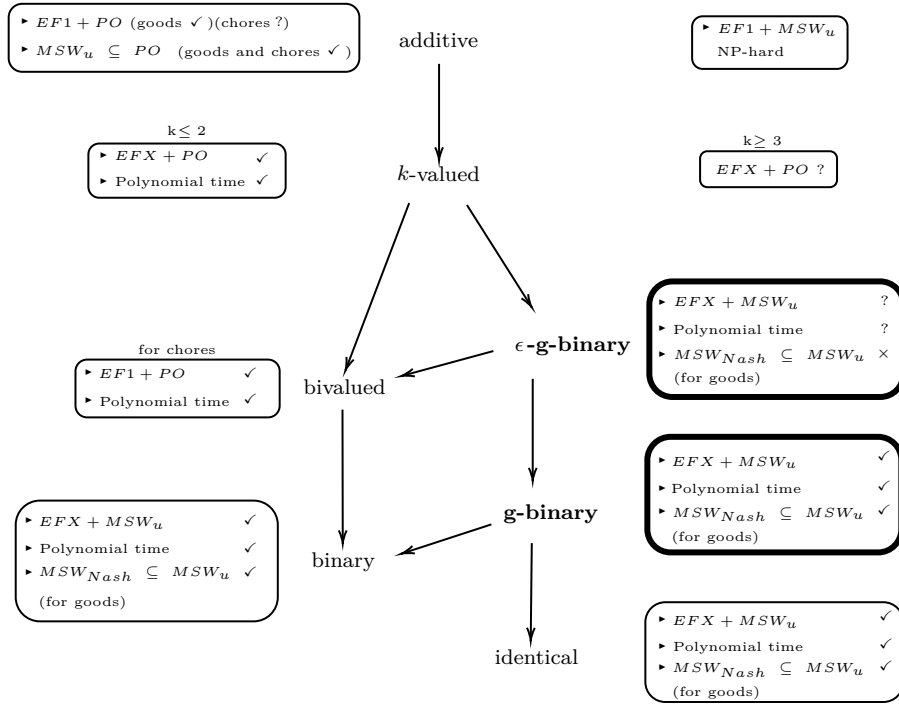


Figure 1: The boxes having a bold face framework represent the classes and results proposed in this work. The arrows show strict inclusion. Check marks (✓) denote the existence of fairness (EFX) and efficiency (PO) properties, while (?) denotes an open question.

## 7. Concluding remarks

In this work, we have studied fair and efficient allocations for indivisible resources (goods and chores) when the valuations are defined through g-binary utility functions. We showed that there exists an allocation that is envy-free up to one item (EF1) (for goods and chores) and maximizes utilitarian social welfare ( $MSW_u$ ) using g-binary utility functions. This result can be slightly modified to produce an allocation that is envy-free up to any item valued different from zero (EFX) (for goods and chores). In this framework a characterization of Pareto optimality is given.

Actually, we build algorithms for finding allocations which are EF1, EFX and simultaneously  $MSW_u$  (for goods and chores). They run in polynomial-time. However, the algorithm finding an allocation EF1 is computationally slightly better than the algorithm finding an allocation EFX.

The results of Section 5 show, in some way, that the g-binary scenarios are a sort of limiting scenario in which it is easy to provide EF1 and PO properties related to utilitarian social welfare. This is achieved by considering when a resource is indifferent to an agent. However, a little bit beyond that, the results do not hold.



It is known that finding allocations producing a maximum of Nash is NP-hard even in identical scenarios (see [5]). Thus, in g-binary scenarios this problem will necessarily be NP-hard. Then, the natural question is if we can adapt the techniques of Barman et al. [5] in order to prove that Algorithm 1 produces a good approximation of a maximum of Nash social welfare in the case of g-binary scenarios.

### Acknowledgements

The first, second and fourth authors thank the Vice Chancellery of Research and Innovation of Yachay Tech University which has partially funded this work through the *Social Welfare and Justice in Decision Making* project code MATH22-08.

The third author has benefited from the support of the AI Chair BE4musIA of the French National Research Agency (ANR-20-CHIA-0028) and has also been partially funded by the program PAUSE of Collège de France.

Thanks to the referees whose observations have greatly helped to improve the results of this work and its presentation. We also thank Professor Olga Porras for her careful proof reading.

### Appendix A. Algorithm, proofs and examples

A more detailed explanation of Algorithm 1 is as follows:

1. Let  $v_0 = (0, \dots, 0)$  be the initial vector of partial valuations, with size  $n$ .
2. For  $k = 1$  to  $m$ :
  - (a) We take  $r_k \in \mathcal{R}$ .
  - (b)  $\alpha_k = \max\{u_j(r_k) : j \in \mathcal{A}\}$ .
  - (c) Let  $P_k$  and  $M_k$  be the sets given by:

$$P_k = \{j \in \mathcal{A} : u_j(r_k) = \alpha_k\} \tag{A.1}$$

and

$$l_k = \min \{|[v_{k-1}]_j| : j \in P_k\}$$

$$M_k = \{i \in P_k : |[v_{k-1}]_i| = l_k\} \tag{A.2}$$

where  $[v_{k-1}]_i$  is the position  $i$  of  $v_{k-1}$ .

- (d) Let  $j_k$  be the minimum of  $M_k$ .
- (e) Allocation of  $r_k$ :

$$\Gamma(r_k) = j_k \tag{A.3}$$

(f) Updating the vector  $v_k$  of partial utilities, for all  $i \in \mathcal{A}$ :

$$[v_k]_i = \begin{cases} [v_{k-1}]_i + u_{j_k}(r_k), & \text{if } i = j_k \\ [v_{k-1}]_i, & \text{if } i \neq j_k. \end{cases}$$

Notice that  $P_k \neq \emptyset$ . Therefore,  $M_k \neq \emptyset$  and  $M_k \subseteq \mathbb{N}$ ; so, there exists a minimum for  $M_k$ . On the other hand,  $\forall i \in \mathcal{A}$ ,  $u_i$  is an additive utility function, so each position  $i$  of  $v_k$ ,  $[v_k]_i$ , is the partial valuation given by agent  $i$  to its assigned bundle up to step  $k$ .

**Theorem 3.** *Assume a goods (chores) g-binary scenario and let  $F$  be an allocation in  $\mathcal{A}^{\mathcal{R}}$ . Then,  $F$  is PO if, and only if,  $F \in MSW_u$ .*

**Proof:** (*only if* part) Let  $F \in \mathcal{A}^{\mathcal{R}}$ , and we suppose that  $F$  does not maximize utilitarian social welfare. We want to show that  $F$  is not Pareto optimal.

Since all agents consider g-binary utility functions and  $F$  does not maximize utilitarian social welfare, by Theorem 5, there exist  $r^* \in \mathcal{R}$  and  $j \in \mathcal{A}$  such that  $u_j(r^*) = \max\{0, p^*\}$  and  $u_i(r^*) = \min\{0, p^*\}$  with  $i = F(r^*)$  and  $p^* \neq 0$ . Let  $G \in \mathcal{A}^{\mathcal{R}}$  be given by

$$G(r) = \begin{cases} F(r), & \text{if } r \neq r^* \\ j, & \text{if } r = r^* \end{cases}$$

Note that  $r^* \notin F^{-1}(j)$ ,  $G^{-1}(j) = F^{-1}(j) \cup \{r^*\}$ ,  $G^{-1}(i) = F^{-1}(i) \setminus \{r^*\}$ , and for all  $k \in \mathcal{A}$  with  $k \notin \{i, j\}$ ,  $F^{-1}(k) = G^{-1}(k)$ .

In a good g-binary scenario,  $u_j(r^*) = p^* > 0$  and  $u_i(r^*) = 0$ . So,  $u_j(F^{-1}(j)) < u_j(G^{-1}(j))$  and  $u_k(F^{-1}(k)) = u_k(G^{-1}(k))$  for  $k \neq j$ . Hence,  $F$  is Pareto dominated by  $G$ .

Now, in a chore g-binary scenario,  $u_j(r^*) = 0$  and  $u_i(r^*) = p^* < 0$ . Thus,  $u_i(G^{-1}(i)) = u_i(F^{-1}(i)) - u_i(r^*) > u_i(F^{-1}(i))$  and  $u_k(F^{-1}(k)) = u_k(G^{-1}(k))$  for  $k \neq j$ . Therefore,  $G$  dominates  $F$ .

In both cases  $F$  is not PO.

(*if* part) Follows from Theorem 1. ■

The following observation is very useful in the proofs. Its proof is obvious by equations (A.3), (A.1) and (A.2).

**Remark 2.** *We assume an additive scenario. Let  $\Gamma$  be the allocation of Algorithm 1. For every  $r_k \in \mathcal{R}$  and every  $j \in \mathcal{A}$  such that  $\Gamma(r_k) = j$ , we have*

1.  $\forall i (i \in P_k \implies |[v_{k-1}]_j| \leq |[v_{k-1}]_i|)$
2.  $u_j(r_k) = \alpha_k = \max\{u_i(r_k) : \forall i \in \mathcal{A}\}$

**Theorem 6.** *Under a goods (chores) additive scenario,  $\Gamma \in MSW_u$  and it is obtained in  $O(nm)$  operations.*

**Proof:** Let  $\Gamma$  be the allocation of Algorithm 1. By Remark 2, part 2, and Theorem 5, we have that  $\Gamma \in MSW_u$ . The proposed algorithm starts

initializing the vector of partial utilities  $v_0$  with zeros, which has one position by each one of the  $n$  agents. This step demands  $O(n)$  operations. Step-2 allocates the  $m$  resources finding the agent with minimum partial utility in each iteration, following (A.2). Allocating  $m$  resources, finding the minimum of  $n$  agents demands  $O(nm)$  operations. In the next steps, the agent receives the resource in (A.3); for all the resources, this step and the updating of the vector of partial utilities runs in  $O(m)$ . Building the resource allocation runs in  $O(\max(n, nm, m))$ , thus, we have that the proposed algorithm is in  $O(nm)$ . ■

In order to prove Theorem 7, we establish a technical lemma for goods or chores g-binary scenario. To establish this lemma we need the following definitions:

$$\begin{aligned} A_{ij} &= \{r \mid u_i(r) = \max\{0, p_r\} \wedge u_j(r) = \min\{0, p_r\}\}, \\ B_{ij} &= \{r \mid u_i(r) = \min\{0, p_r\} \wedge u_j(r) = \max\{0, p_r\}\}, \\ C_{ij} &= \{r \mid u_i(r) = \max\{0, p_r\} = u_j(r)\}, \\ D_{ij} &= \{r \mid u_i(r) = \min\{0, p_r\} = u_j(r)\}, \\ R_* &= \{r \mid u_i(r) = \min\{0, p_r\}, \forall i \in A\}. \end{aligned}$$

Note that for every  $i, j$  we have

$$R = A_{ij} \cup B_{ij} \cup C_{ij} \cup D_{ij} \quad (\text{A.4})$$

and this is a partition of  $R$ .

**Lemma 1.** *Under a g-binary scenario for goods or chores, if  $\Gamma$  is computed via Algorithm 1, then for all  $i, j \in A$  we have that*

$$u_i(\Gamma^{-1}(i)) = \begin{cases} u_i(\Gamma^{-1}(i) \cap A_{ij}) + u_i(\Gamma^{-1}(i) \cap C_{ij}) & \text{in goods g-binary scenario} \\ u_i(\Gamma^{-1}(i) \cap R_*) & \text{in chores g-binary scenario} \end{cases} \quad (\text{A.5})$$

and,

$$u_i(\Gamma^{-1}(j)) = \begin{cases} u_i(\Gamma^{-1}(j) \cap C_{ij}) & \text{in goods g-binary scenario} \\ u_i(\Gamma^{-1}(j) \cap B_{ij}) + u_i(\Gamma^{-1}(j) \cap R_*) & \text{in chores g-binary scenario} \end{cases} \quad (\text{A.6})$$

**Proof:** For each  $r \in R$ , we consider  $M_r = \{k \in A : u_k(r) = \max\{0, p_r\}\}$  and  $m_r = \{k \in A : u_k(r) = \min\{0, p_r\}\}$ . Note that,  $M_r \cap m_r = \emptyset$  and  $M_r \cup m_r = A$ . Moreover, by definition of  $\Gamma$ , if  $M_r \neq \emptyset$ , then  $\Gamma(r) \in M_r$ .

**Claim 1.** *For all  $i, j \in A$ ,*

1.  $A_{ij} \cap \Gamma^{-1}(j) = \emptyset = B_{ij} \cap \Gamma^{-1}(i)$ .  
*In fact, if  $r \in A_{ij} \cap \Gamma^{-1}(j)$ , then  $\Gamma(r) = j \in M_r$  and  $j \in m_r$  which is a contradiction. So,  $A_{ij} \cap \Gamma^{-1}(j) = \emptyset$ . As  $B_{ij} = A_{ji}$ , then  $B_{ij} \cap \Gamma^{-1}(i) = \emptyset$ .*

2.  $D_{ij} \cap \Gamma^{-1}(i) = R_* \cap \Gamma^{-1}(i)$ .

Suppose that  $R_* = \emptyset$ . Then  $M_r \neq \emptyset$  for all  $r$ . If  $r \in D_{ij} \cap \Gamma^{-1}(i)$ , then  $\Gamma(r) = i \in M_r$  and  $i \in m_r$ , which is a contradiction. So,  $D_{ij} \cap \Gamma^{-1}(i) = \emptyset = R_* \cap \Gamma^{-1}(i)$ . Now, suppose that  $R_* \neq \emptyset$ , if  $r \in R_*$ , then  $m_r \neq \emptyset$  and, by definition of  $\Gamma$ ,  $\Gamma(r) \in m_r$ . Given  $r \in D_{ij} \cap \Gamma^{-1}(i)$ , then  $u_i(r) = \min\{0, p_r\} = u_j(r)$  and  $\Gamma(r) = i$ . Therefore, for all  $k \in A$ ,  $u_k(r) = \min\{0, p_r\}$ ; otherwise,  $\Gamma(r) \notin m_r$ . Thus,  $r \in R_* \cap \Gamma^{-1}(i)$ . But,  $R_* \subseteq D_{ij}$ . Thus,  $D_{ij} \cap \Gamma^{-1}(i) = R_* \cap \Gamma^{-1}(i)$ .

3.  $\Gamma^{-1}(i) = (A_{ij} \cap \Gamma^{-1}(i)) \cup (C_{ij} \cap \Gamma^{-1}(i)) \cup (R_* \cap \Gamma^{-1}(i))$ .

This is due to parts 1 and 2 of Claim1 and Equation A.4.

Now we prove the lemma. By Claim1 part 3 and the fact that  $B_{ij} = A_{ji}$  and  $C_{ij} = C_{ji}$ , we have

$$\begin{aligned} u_i(\Gamma(i)) &= u_i(A_{ij} \cap \Gamma^{-1}(i)) + u_i(C_{ij} \cap \Gamma^{-1}(i)) + u_i(R_* \cap \Gamma^{-1}(i)) \\ u_i(\Gamma(j)) &= u_i(B_{ij} \cap \Gamma^{-1}(j)) + u_i(C_{ij} \cap \Gamma^{-1}(j)) + u_i(R_* \cap \Gamma^{-1}(j)) \end{aligned}$$

If the scenario is a chore g-binary scenario, then  $u_i(A_{ij} \cap \Gamma^{-1}(i)) = u_i(C_{ij} \cap \Gamma^{-1}(i)) = u_i(C_{ij} \cap \Gamma^{-1}(j)) = 0$ . While in a good g-binary scenario,  $u_i(R_* \cap \Gamma^{-1}(i)) = u_i(R_* \cap \Gamma^{-1}(j)) = u_i(B_{ij} \cap \Gamma^{-1}(j)) = 0$ . Thus, the equations (A.5) and (A.6) are true.  $\blacksquare$

**Theorem 7.** *Under a g-binary scenario for goods or chores, the allocation  $\Gamma$  given by Algorithm 1 produces a maximal utilitarian welfare, is EF1 and PO. Its run time is  $O(mn)$ .*

**Proof:** We prove first the case of goods. We suppose that  $\forall i \in \mathcal{A}$ ,  $u_i$  is a g-binary utility function. We consider the allocation  $\Gamma$  of Algorithm 1. By Theorem 6,  $\Gamma$  maximizes utilitarian social welfare in  $O(mn)$  operations, and by Theorem 3,  $\Gamma$  is PO. We want to show that  $\Gamma$  is EF1. Suppose that there exists  $i \in A$  such that

$$u_i(\Gamma^{-1}(i)) < u_i(\Gamma^{-1}(j)) \quad (\text{A.7})$$

for some agent  $j$ . As  $u_i$  is a g-binary utility, by Lemma 1, it is enough to show that there exists  $r \in \Gamma^{-1}(j) \cap C_{ij}$  such that  $u_i(\Gamma^{-1}(i)) \geq u_i(\Gamma^{-1}(j)) - u_i(r)$ . Clearly,  $\Gamma^{-1}(j) \cap C_{ij} \neq \emptyset$ . Otherwise, by (A.6) and (A.7), we have  $u_i(\Gamma^{-1}(j)) < 0$ , which is a contradiction because  $u_i$  is non negative. Let  $\Gamma^{-1}(j) \cap C_{ij} = \{r_{s_1}, \dots, r_{s_k}\}$ , where  $r_{s_k}$  is the last resource assigned to  $j$  and it is preferred by  $i$  and  $j$ . Since  $\Gamma(r_{s_k}) = j$  and  $i \in P_{s_k}$ , by Remark 2, part 1,

$$u_i(\Gamma^{-1}(i) \cap \{r_1, \dots, r_{s_k-1}\}) \geq u_j(\Gamma^{-1}(j) \cap \{r_1, \dots, r_{s_k-1}\}).$$

By additivity of  $u_j$ ,

$$u_j(\Gamma^{-1}(j) \cap \{r_1, \dots, r_{s_k-1}\}) \geq u_j(\Gamma^{-1}(j) \cap \{r_1, \dots, r_{s_k-1}\} \cap C_{ij}).$$

Using the transitivity of  $\geq$ , we have

$$u_i(\Gamma^{-1}(i) \cap \{r_1, \dots, r_{s_k-1}\}) \geq u_j(\Gamma^{-1}(j) \cap \{r_1, \dots, r_{s_k-1}\} \cap C_{ij}). \quad (\text{A.8})$$

As  $u_i$  and  $u_j$  are utility functions in a g-binary scenario, then the utility that agents  $i$  and  $j$  assign to each resource in  $C_{ij}$  is equal, so

$$u_j(\Gamma^{-1}(j) \cap \{r_1, \dots, r_{s_k-1}\} \cap C_{ij}) = u_i(\Gamma^{-1}(j) \cap \{r_1, \dots, r_{s_k-1}\} \cap C_{ij}). \quad (\text{A.9})$$

By equations (A.8) and (A.9),

$$u_i(\Gamma^{-1}(i) \cap \{r_1, \dots, r_{s_k-1}\}) \geq u_i(\Gamma^{-1}(j) \cap \{r_1, \dots, r_{s_k-1}\} \cap C_{ij}). \quad (\text{A.10})$$

Now, by equation (A.6) and additivity of  $u_i$ ,

$$\begin{aligned} u_i(\Gamma^{-1}(j)) &= u_i(\Gamma^{-1}(j) \cap C_{ij}) \\ &= u_i(\Gamma^{-1}(j) \cap \{r_1, \dots, r_{s_k-1}\} \cap C_{ij}) + u_i(r_{s_k}) \end{aligned}$$

and, by equation (A.10),

$$u_i(\Gamma^{-1}(j)) \leq u_i(\Gamma^{-1}(i) \cap \{r_1, \dots, r_{s_k-1}\}) + u_i(r_{s_k});$$

so,

$$u_i(\Gamma^{-1}(i) \cap \{r_1, \dots, r_{s_k-1}\}) \geq u_i(\Gamma^{-1}(j)) - u_i(r_{s_k}).$$

But,  $u_i(\Gamma^{-1}(i)) \geq u_i(\Gamma^{-1}(i) \cap \{r_1, \dots, r_{s_k-1}\})$ , and using transitivity,

$$u_i(\Gamma^{-1}(i)) \geq u_i(\Gamma^{-1}(j)) - u_i(r_{s_k}). \quad (\text{A.11})$$

Therefore,  $\Gamma$  is EF1.

Now we give the proof in the case of chores. By Lemma 1, for all  $i, j \in A$ ,

$$u_i(\Gamma^{-1}(i)) = u_i(\Gamma^{-1}(i) \cap R_*) \quad (\text{A.12})$$

and

$$u_i(\Gamma^{-1}(j)) = u_i(\Gamma^{-1}(j) \cap B_{ij}) + u_i(\Gamma^{-1}(j) \cap R_*). \quad (\text{A.13})$$

If  $R_* = \emptyset$ , then  $u_i(\Gamma^{-1}(i)) = 0$  and  $u_i(\Gamma^{-1}(j)) = u_i(\Gamma^{-1}(j) \cap B_{ij}) \leq 0$ . Thus,  $u_i(\Gamma^{-1}(i)) \geq u_i(\Gamma^{-1}(j))$ . So,  $\Gamma$  is EF. In consequence,  $\Gamma$  is EF1.

Suppose that  $R_* \neq \emptyset$  and that there exist  $i, j \in A$  such that

$$u_i(\Gamma^{-1}(i)) < u_i(\Gamma^{-1}(j)). \quad (\text{A.14})$$

We want to show that there is  $r \in \Gamma^{-1}(i)$  such that

$$u_i(\Gamma^{-1}(i) \setminus \{r\}) \geq u_i(\Gamma^{-1}(j)) \quad (\text{A.15})$$

By equation (A.14),  $u_i(\Gamma^{-1}(i)) < 0$ ; and by equation (A.12),  $\Gamma^{-1}(i) \cap R_* \neq \emptyset$ . Using additivity of  $u_i$  and equations (A.12) and (A.13), to show that (A.15) is true, it is enough to find  $r \in \Gamma^{-1}(i) \cap R_*$  such that

$$u_i((\Gamma^{-1}(i) \cap R_*) \setminus \{r\}) \geq u_i(\Gamma^{-1}(j)). \quad (\text{A.16})$$

Let  $r_s \in \Gamma^{-1}(i) \cap R_*$  such that  $r_s$  is the last chore allocated to  $i$  and it is minimized for all  $l \in A$ . Consider  $\{r_1, \dots, r_{s-1}\}$ <sup>5</sup> the set of all chores allocated before  $r_s$ . As

$$\{r_1, \dots, r_{s-1}\} \cap \Gamma^{-1}(l) \cap R_* \subseteq \Gamma^{-1}(l)$$

then

$$u_i(R_* \cap \{r_1, \dots, r_{s-1}\} \cap \Gamma^{-1}(l)) \geq u_i(\Gamma^{-1}(l)). \quad (\text{A.17})$$

On the other hand, as  $r_s$  is the last chore assigned to  $i$  and  $r_s \in R_*$ , then

$$u_i(R_* \cap \Gamma^{-1}(i) \setminus \{r_s\}) = u_i(R_* \cap \{r_1, \dots, r_{s-1}\} \cap \Gamma^{-1}(i))$$

and by Lemma 1,

$$u_i(R_* \cap \{r_1, \dots, r_{s-1}\} \cap \Gamma^{-1}(i)) = u_i(\{r_1, \dots, r_{s-1}\} \cap \Gamma^{-1}(i))$$

then, by transitivity,

$$u_i(R_* \cap \Gamma^{-1}(i) \setminus \{r_s\}) = u_i(\{r_1, \dots, r_{s-1}\} \cap \Gamma^{-1}(i)). \quad (\text{A.18})$$

In order to complete the proof, we only have to show that

$$u_i(\{r_1, \dots, r_{s-1}\} \cap \Gamma^{-1}(i)) \geq u_i(R_* \cap \{r_1, \dots, r_{s-1}\} \cap \Gamma^{-1}(j)) \quad (\text{A.19})$$

because if in (A.17), we take  $l = j$  and use transitivity between (A.18), (A.19) and (A.17), the expression (A.16) is true.

As  $\Gamma(r_s) = i$ , by the way the  $\Gamma$  is defined, then  $\forall l \in A$

$$u_i(\{r_1, \dots, r_{s-1}\} \cap \Gamma^{-1}(i)) \geq u_l(\{r_1, \dots, r_{s-1}\} \cap \Gamma^{-1}(l)).$$

But, for all  $l \in A$ ,

$$\begin{aligned} u_l(\{r_1, \dots, r_{s-1}\} \cap \Gamma^{-1}(l)) &= u_l(R_* \cap \{r_1, \dots, r_{s-1}\} \cap \Gamma^{-1}(l)) \\ &= u_i(R_* \cap \{r_1, \dots, r_{s-1}\} \cap \Gamma^{-1}(l)) \end{aligned}$$

and

$$u_i(R_* \cap \{r_1, \dots, r_{s-1}\} \cap \Gamma^{-1}(l)) = u_i(R_* \cap \{r_1, \dots, r_{s-1}\} \cap \Gamma^{-1}(l)).$$

Using transitivity, we get (A.19). ■

**Theorem 8.** *Under a  $g$ -binary scenario for goods or chores, the allocation  $\Gamma^*$  given by Algorithm 1 modified as previously indicated maximizes the utilitarian social welfare, is EFX and PO. Moreover, its run time is  $O(m \log m + mn)$ .*

**Proof:** The argument of the proof is similar to that in the proof of the Theorem 7. Let  $i, j$  be in  $\mathcal{A}$  such that

$$u_i(\Gamma^{*-1}(i)) < u_i(\Gamma^{*-1}(j)).$$

---

<sup>5</sup>Note that if  $r_s = r_1$ , then  $\{r_1, \dots, r_{s-1}\} = \emptyset$

First we consider the case of goods. By Lemma 1,

$$u_i(\Gamma^{*-1}(j)) = u_i(\Gamma^{*-1}(j) \cap C_{ij}) = \{r_{s_1}, \dots, r_{s_k}\}.$$

As  $\mathcal{R}$  is sorted in descending order, then for all  $r \in \Gamma^{*-1}(j)$  with  $u_i(r) > 0$ ,

$$u_i(r_{s_k}) \leq u_i(r).$$

By equation (A.11)

$$u_i(\Gamma^{*-1}(i)) \geq u_i(\Gamma^{*-1}(j)) - u_i(r_{s_k}).$$

So, for all  $r \in \Gamma^{*-1}(j)$  with  $u_i(r) > 0$ ,

$$u_i(\Gamma^{*-1}(i)) \geq u_i(\Gamma^{*-1}(j)) - u_i(r).$$

Thus,  $\Gamma^*$  is *EFX*.

Now, we consider the case of chores. By Lemma 1,

$$u_i(\Gamma^{*-1}(i)) = u_i(\Gamma^{*-1}(i) \cap \mathcal{R}_*) = u_i(\Gamma^{*-1}(i) \cap \mathcal{R}_* \cap \{r_1, \dots, r_s\})$$

where  $r_s$  is the last chore in  $\mathcal{R}_*$  allocated to  $i$ . Note that,  $\forall r \in \Gamma^{*-1}(i) \setminus \mathcal{R}_*$ ,  $u_i(r) = 0$ . As  $\mathcal{R}$  is sorted in increasing order then for all  $r \in \Gamma^{*-1}(i)$  with  $u_i(r) < 0$ ,

$$u_i(r_s) \geq u_i(r).$$

Then

$$u_i(\Gamma^{*-1}(i)) - u_i(r) \geq u_i(\Gamma^{*-1}(i)) - u_i(r_s). \quad (\text{A.20})$$

Since (A.16) is true for  $r_s$ , using Transitivity, from (A.20) and (A.16), we have that  $\forall r \in \Gamma^{*-1}(i)$  with  $u_i(r) < 0$ ,

$$u_i(\Gamma^{*-1}(i)) - u_i(r) \geq u_i(\Gamma^{*-1}(j)).$$

Thus,  $\Gamma^*$  is *EFX*. ■

The following example shows the facts established in Remark 1 about the behavior of  $\Gamma$  and other allocations in a g-binary scenario.

**Example 5.** Suppose that  $\mathcal{R} = \{r_1, r_2, r_3, r_4, r_5\}$  and that each resource  $r_k$  is valued as  $p_k$  according to Table A.11.

Table A.11:  $p_k$  values for each resource.

	$r_1$	$r_2$	$r_3$	$r_4$	$r_5$
$p_k$	500	200	50	100	250

Let  $\mathcal{A} = \{1, 2, 3\}$  be the set of agents; each agent  $i$  establishes its utility functions  $u_i$  over each resource using Table A.12.

Table A.12: Utility functions.

	$r_1$	$r_2$	$r_3$	$r_4$	$r_5$
$u_1$	500	200	50	0	0
$u_2$	500	0	50	100	250
$u_3$	500	200	0	100	0

Note that all agents prefer the resource  $r_1$  and their utility is 500. The resource  $r_2$  is required by agents 1 and 3, and their utility is 200; the agents 1 and 2 give to resource  $r_3$  the utility of 50. Concerning the resource  $r_4$ , agents 2 and 3 give it the utility of 100; whereas resource  $r_5$  is only required by agent 2 with utility 250. If for  $i = 1, 2, 3$ , the function  $u_i$  is extended additively over each subset  $S \subseteq \mathcal{R}$ , that is,  $\forall i \in \mathcal{A}$ ,  $u_i(S) = \sum_{r \in S} u_i(\{r\})$ , then each  $u_i$  is a  $g$ -binary utility function, i.e., it is a  $g$ -binary scenario.

In this scenario, we will consider four allocations in order to illustrate their behavior with respect to properties  $EF1$ ,  $PO$ ,  $MSW_u$  and  $MSW_{Nash}$ . Let  $F$ ,  $G$ ,  $\Gamma$  and  $J$  be the allocations defined by Table A.13, which shows the agent number who receives each resource.

Table A.13: Allocations.

	$r_1$	$r_2$	$r_3$	$r_4$	$r_5$
$F$	1	3	3	3	2
$G$	1	3	2	3	2
$\Gamma$	1	3	2	2	2
$J$	1	1	2	3	2

In Table A.14, we show the utility assigned by each agent to its received bundle, and the social welfare in each allocation.

Table A.14: Utility by received bundle and social welfare.

	1	2	3	$SW_u$	$SW_{Nash}$
$u_i(F^{-1}(i))$	500	250	300	1050	37500000
$u_i(G^{-1}(i))$	500	300	300	1100	45000000
$u_i(\Gamma^{-1}(i))$	500	400	200	1100	40000000
$u_i(J^{-1}(i))$	700	300	100	1100	21000000

The properties of each allocation are described in Table A.15 and the detailed verification of these facts is given below.



Table A.15: Allocation properties under a g-binary scenario.

	PO	EF1	$MSW_u$	$MSW_{Nash}$
$F$	×	✓	×	×
$G$	✓	✓	✓	✓
$\Gamma$	✓	✓	✓	×
$J$	✓	×	✓	×

It is important to note that  $F$  is an allocation EF1 and it is not in  $MSW_u$ .  $G$  satisfies all the properties.  $\Gamma$  is EF1, PO, it is in  $MSW_u$  but it is not in  $MSW_{Nash}$ . Finally,  $J$  is PO and it is in  $MSW_u$  but it is neither EF1 nor in  $MSW_{Nash}$ .

**Proof of Properties in Example 5:** We give details of the properties fulfilled by the allocations in Table A.15.

From Theorem 5 and Table A.11, it is easy to see that the maximum utilitarian welfare is reached in 1100. Moreover;  $G$ ,  $\Gamma$  and  $J$  maximize  $SW_u$ . By Theorem 1 and Table A.14, we have that  $G$ ,  $\Gamma$ , and  $J$  are PO. On the other hand, from Table A.14, we can observe that agents 1 and 3 in the allocations  $F$  and  $G$ , have the same utility for the received bundle; but, in  $G$ , agent 2 improves its utility. Then,  $F$  is Pareto dominated by  $G$ . Therefore,  $F$  is not PO.

A search determined that the maximum Nash social welfare is reached at 45000000. Then  $G$  is a maximum Nash social welfare and, by Theorem 2,  $G$  is EF1. Moreover, allocations  $F$  and  $\Gamma$  are EF1. In fact,  $\Gamma$  is EF1 by Theorem 6.

For  $F$ , agents 2 and 3 envy agent 1, however, the envy disappears when eliminating  $r_1$ . Finally,  $J$  is not EF1, because agent 3 envies agent 1,  $u_3(J^{-1}(3)) < u_3(J^{-1}(1))$  and

$$u_3(J^{-1}(3)) = 100 < 200 = u_3(J^{-1}(1) \setminus \{r_1\}) < 500 = u_3(J^{-1}(1) \setminus \{r_2\}).$$

■

The following example shows that  $\Gamma^*$  is not EFX<sub>0</sub>.

**Example 6.** Let's consider the following g-binary scenario where  $n = 3$ ,  $m = 8$ . Table A.16 shows the utility functions that each agent gives to each resource.

Table A.16: Utility functions.

	$r_1$	$r_2$	$r_3$	$r_4$	$r_5$	$r_6$	$r_7$	$r_8$
$u_1$	20	0	10	2	0	0	3	1
$u_2$	20	0	10	2	11	19	0	1
$u_3$	20	9	0	2	0	19	3	1

Now, using the modified Algorithm 1, we get the following allocation  $\Gamma^*$ .

Table A.17: Allocation  $\Gamma^*$ 

	$r_1$	$r_2$	$r_3$	$r_4$	$r_5$	$r_6$	$r_7$	$r_8$
$\Gamma^*$	1	3	1	3	2	2	3	3

Then, in Table A.18 we show the utility assigned by each agent to its received bundle, and the social welfare in each allocation.

Table A.18: Utility by received bundle and social welfare.

	1	2	3	$SW_U$	$SW_{Nash}$
$u_i(\Gamma^{*-1}(i))$	30	30	15	75	13500

Notice that agent 3 envies agent 2 because

$$u_3(\{r_2, r_5, r_6\}) = 15$$

$$u_3(\{r_5, r_6\}) = 19$$

which means that

$$u_3(\{r_2, r_5, r_6\}) < u_3(\{r_5, r_6\}).$$

Since  $\Gamma^*$  was obtained using the modified Algorithm 1, we obtain that  $\Gamma^*$  is EFX. Now we check that  $\Gamma^*$  does not satisfy EFX<sub>0</sub>. Observe that

$$u_3(\{r_5, r_6\} \setminus \{r_5\}) = 19 \quad \text{given that} \quad u_3(\{r_5\}) = 0$$

so, it follows that

$$u_3(\{r_2, r_5, r_6\}) = 15 < 19 = u_3(\{r_5, r_6\} \setminus \{r_5\}) \quad (\text{A.21})$$

Thus, property EFX<sub>0</sub> is not satisfied.

## References

- [1] AMANATIDIS, G., BIRMPAS, G., FILOS-RATSIKAS, A., HOLLENDER, A., AND VOUDOURIS, A. A. Maximum nash welfare and other stories about efx. *Theoretical Computer Science* 863 (2021), 69–85.
- [2] AZIZ, H., BIRÓ, P., LANG, J., LESCA, J., AND MONNOT, J. Optimal reallocation under additive and ordinal preferences. In *Proceedings of the 2016 International Conference on Autonomous Agents & Multiagent Systems* (Richland, SC, 2016), AAMAS '16, International Foundation for Autonomous Agents and Multiagent Systems, p. 402–410.
- [3] AZIZ, H., CARAGIANNIS, I., IGARASHI, A., AND WALSH, T. Fair allocation of combinations of indivisible goods and chores, 2018. arXiv:1807.10684 [cs.GT].

- [4] AZIZ, H., HUANG, X., MATTEI, N., AND SEGAL-HALEVI, E. Computing welfare-maximizing fair allocations of indivisible goods, 2021. arXiv:2012.03979 [cs:GT].
- [5] BARMAN, S., KRISHNAMURTHY, S. K., AND VAISH, R. Greedy algorithms for maximizing nash social welfare. In *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems* (Richland, SC, 2018), AAMAS '18, International Foundation for Autonomous Agents and Multiagent Systems, p. 7–13.
- [6] BRAINARD, W. C., AND SCARF, H. E. How to compute equilibrium prices in 1891. *American Journal of Economics and Sociology* 64, 1 (2005), 57–83.
- [7] BUDISH, E. The combinatorial assignment problem: Approximate competitive equilibrium from equal incomes. *Journal of Political Economy* 119, 6 (2011), 1061–1103.
- [8] CAMACHO, F., ZHUNIO, C., FONSECA, R., AND CASTILLO, Z. Asignación de recursos con eficiencia y justicia débil en el bienestar social utilitario. *Revista Ibérica de Sistemas e Tecnologías de Informação* E32 (2020), 583–595.
- [9] CARAGIANNIS, I., KUROKAWA, D., MOULIN, H., PROCACCIA, A. D., SHAH, N., AND WANG, J. The unreasonable fairness of maximum nash welfare. *ACM Trans. Econ. Comput.* 7, 3 (Sept. 2019).
- [10] CHEVALEYRE, Y., ENDRISS, U., AND MAUDET, N. Distributed fair allocation of indivisible goods. *Artificial Intelligence* 242 (2017), 1–22.
- [11] CONITZER, V., FREEMAN, R., AND SHAH, N. Fair public decision making. In *Proceedings of the 2017 ACM Conference on Economics and Computation* (New York, NY, USA, 2017), EC '17, Association for Computing Machinery, p. 629–646.
- [12] DE KEIJZER, B., BOUVERET, S., KLOS, T., AND ZHANG, Y. On the complexity of efficiency and envy-freeness in fair division of indivisible goods with additive preferences. In *Algorithmic Decision Theory* (Berlin, Heidelberg, 2009), F. Rossi and A. Tsoukias, Eds., Springer Berlin Heidelberg, pp. 98–110.
- [13] EBADIAN, S., PETERS, D., AND SHAH, N. How to fairly allocate easy and difficult chores. *arXiv:2110.11285* (2021).
- [14] EBADIAN, S., PETERS, D., AND SHAH, N. How to fairly allocate easy and difficult chores. In *21st International Conference on Autonomous Agents and Multiagent Systems, AAMAS 2022, Auckland, New Zealand, May 9-13, 2022* (2022), P. Faliszewski, V. Mascardi, C. Pelachaud, and M. E. Taylor, Eds., International Foundation for Autonomous Agents and Multiagent Systems (IFAAMAS), pp. 372–380.

- [15] NASH, J. F. The bargaining problem. *Econometrica* 18, 2 (1950), 155–162.
- [16] RAMEZANI, S., AND ENDRISS, U. Nash social welfare in multiagent resource allocation. In *Agent-Mediated Electronic Commerce. Designing Trading Strategies and Mechanisms for Electronic Markets* (Berlin, Heidelberg, 2010), E. David, E. Gerding, D. Sarne, and O. Shehory, Eds., Springer Berlin Heidelberg, pp. 117–131.
- [17] STEINHAUS, H. The problem of fair division. *Econometrica* 16 (1948), 101–104.
- [18] STEINHAUS, H. Sur la division pragmatique. *Econometrica* 17 (1949), 315–319.