On the Complexity of Efficiency and Envy-Freeness in Fair Division of Indivisible Goods with Additive Preferences

Bart de Keijzer[1], Sylvain Bouveret[2], Tomas Klos[1], and Yingqian Zhang[1]

[1] Delft University of Technology
B.deKeijzer@student.tudelft.nl, {T.B.Klos,Yingqian.Zhang}@tudelft.nl
[2] Onera-DTIM, Toulouse
Sylvain.Bouveret@onera.fr

Abstract. We study the problem of allocating a set of indivisible goods to a set of agents having additive preferences. We introduce two new important complexity results concerning efficiency and fairness in resource allocation problems: we prove that the problem of deciding whether a given allocation is Pareto-optimal is coNP-complete, and that the problem of deciding whether there is a Pareto-efficient and envy-free allocation is Σ_2^p-complete.

1 Introduction

The problem of allocating a set of indivisible goods to a set of agents arises in a wide range of applications including, among others, auctions, divorce settlements, frequency allocation, airport traffic management, fair and efficient exploitation of Earth Observation Satellites [1]. In many such real-world problems, one needs to find *efficient* and *fair* solutions, where an efficient solution can be seen informally as ensuring the greatest possible satisfaction to the agents, and where fairness refers to the need for compromises between the agents' (often antagonistic) objectives.

In this paper, we study the resource allocation problem from the point of view of computational complexity. We restrict our setting to *additive* preferences. In other words, the preferences of each agent are represented by a set of weights $w(o)$, standing for the utility (or satisfaction) she enjoys for each single object o. The utility of an agent for a subset of objects S is then given by the sum of the weights of all the objects o in S.

Moreover, we restrict our study to two particular definitions of efficiency and fairness: *Pareto-efficiency* (or Pareto-optimality) and *envy-freeness*. Pareto-efficient allocations are such that we cannot increase the satisfaction of an agent without strictly decreasing the satisfaction of another agent. An allocation is envy-free if and only if each agent likes her share at least as much as the share of any other agent.

In this paper, we introduce two new complexity results concerning the resource allocation problem with additive preferences. Even if the setting seems

F. Rossi and A. Tsoukis (Eds.): ADT 2009, LNAI 5783, pp. 98–110, 2009.

restrictive, we advocate that the particular problems we address are important enough to justify an extensive study for the following reasons. Firstly, one of the most natural ways of (compactly) modeling cardinal preferences over sets of objects (or more generally over combinatorial domains) is to suppose that they are additive. Notice that this goes far beyond resource allocation: matching problems, weighted path in a graph, valued constraints satisfaction problems, etc. Secondly, Pareto-efficiency is one the most prominent notion of efficiency used in collective decision making problems. Thirdly, envy-freeness is a key concept in the literature about resource allocation (see *e.g.* [2]), as it provides an elegant way of encoding the notion of fairness and does not require, contrary to Rawlsian egalitarianism, the interpersonal comparison of utilities.

This paper contributes to fill a gap. On the one hand resource allocation with additive preferences have been extensively studied in economics[1] (see *e.g.* [2,3]), but computational issues (and *a fortiori* complexity) have rarely been considered. On the other hand, computational issues in resource allocation with additive preferences have been studied extensively in computer science (see e.g. [4,5,6]). However, these works mainly concern the optimization of the system's performance as a whole. The properties of Pareto-efficiency and fairness are rarely addressed. Two notable exceptions are the work from Lipton *et al.* [7] that studies envy-freeness in fair resource allocation problems mainly from an algorithmic point of view, and the work from Bouveret and Lang [8] that introduces complexity results for fair resource allocation problems under different hypotheses, including additive preferences. In the latter paper, one result of importance misses, though being conjectured: the complexity of the problem of deciding whether there is a Pareto-efficient and envy-free allocation in a resource allocation problem with additive preferences. This is one of the two main complexity results introduced in our paper and is studied in section 4. The other main complexity result is about the related problem of deciding whether a given allocation is Pareto-efficient when agents have additive preferences. This result is more easily obtained, and is explained in section 3.

2 Background and Notations

In what follows, we will write vectors using arrowed letters (*e.g.* $\overrightarrow{v}$), or brackets for their explicit representations (*e.g.* $\langle v_1, \ldots, v_n \rangle$). v_i will denote the i^{th} component of a vector. Moreover, for any finite set X, $|X|$ will denote the cardinal of X.

In a resource allocation problem, a set of resources must be divided among a set of agents. Since we will focus on additive utility functions only, it suffices to use the following definition of a resource allocation instance:

Definition 1 (Resource allocation instance). *A resource allocation problem is a triple* $\mathcal{P} = \langle A, O, w \rangle$*, where* A *is a set of* agents*,* O *is a set of* indivisible items*, and* $w : A \times O \to \mathbb{R}$ *is a* weight function.

[1] In most social choice studies, utilities stand for amounts of money. Thus additivity is a very natural assumption in this framework.

We define an allocation as follows:

Definition 2 (Allocation). *An allocation for $\mathcal{P} = \langle A, O, w\rangle$ is a vector $\overrightarrow{\pi} = \langle \pi_1, \ldots, \pi_n\rangle \in (2^O)^n$ such that for all $i, j \in A$, $i \neq j \Rightarrow \pi_i \cap \pi_j = \emptyset$. If for every $o \in O$ there exists an i such that $o \in \pi_i$ then $\overrightarrow{\pi}$ is a* complete allocation*.*

Thus, in the problems that we will focus on, the items are non-sharable.

Definition 3 (Individual utility, utility profile). *Let $\mathcal{P} = \langle A, O, w\rangle$ be a resource allocation instance. For all $i \in A$ and $\pi_i \subseteq O$, $u_i(\pi_i) = \sum_{o\in\pi_i} w(i, o)$ is agent i's* individual utility *regarding π_i. Given an allocation $\overrightarrow{\pi}$, the vector $\langle u_1(\pi_1), \ldots, u_n(\pi_n)\rangle$ is the* utility profile *associated to $\overrightarrow{\pi}$.*

Two properties that we will focus on, are *Pareto-efficiency* and *envy-freeness.*

Definition 4 (Pareto-efficiency). *Let $\overrightarrow{\pi}, \overrightarrow{\pi}'$ be two allocations. $\overrightarrow{\pi}$* Pareto-dominates *$\overrightarrow{\pi}'$ if and only if (a) for all i, $u_i(\pi_i) \geq u_i(\pi_i')$, and (b) there exists an i such that $u_i(\pi_i) > u_i(\pi_i')$. $\overrightarrow{\pi}$ is* (Pareto-)efficient *(or* Pareto-optimal*) if and only if there is no $\overrightarrow{\pi}'$ such that $\overrightarrow{\pi}'$ Pareto-dominates $\overrightarrow{\pi}$.*

Definition 5 (Envy & envy-freeness). *We say that an agent $i \in A$ envies another agent $j \in A$ iff $u_i(\pi_j) > u_i(\pi_i)$. An allocation $\overrightarrow{\pi}$ is* envy-free *if and only if $u_i(\pi_i) \geq u_i(\pi_j)$ holds for all i and $j \neq i$.*

In this paper, we will refer to some complexity classes located in the polynomial hierarchy. We assume that the reader is familiar with the complexity class NP and its complementary class coNP. $\Sigma_2^p = \mathsf{NP}^{\mathsf{NP}}$ is the class of all languages recognizable by a nondeterministic Turing machine working in polynomial time using NP oracles. Its complementary class is denoted by Π_2^p.

3 Complexity of Deciding Pareto-optimal Allocations for Agents with Additive Utility

In this section we prove that it is coNP-complete to decide whether an allocation of resources is Pareto-optimal if the agents have additive utility functions. coNP-completeness has already been proved for a generalized case where agents express their utilities explicitly for each bundle of items [9]. coNP-completeness is also known for the case where the agents have k-additive utility functions and $k \geq 2$ [10].[2] This is not explicitly stated in [10], but it follows directly from their proof that it is NP-complete to decide whether it is possible to increase the utilitarian collective utility (*i.e.* sum of individual utilities) of a given allocation, when the agents have 2-additive utility functions. In addition to [10], the problem of maximizing utilitarian collective utility is also explored in [11].

[2] Informally, an agent has k-additive utility if she has a coefficient associated for every set of k items, and her individual utility is the sum of all coefficients associated to the sets of k items that she gets.

The problem we deal with is the following.

Problem 1. Pareto-optimality with additive utility functions (PO-ADD)	
INSTANCE:	A resource allocation instance $\mathcal{P} = \langle A, O, w\rangle$, an allocation $\overrightarrow{\pi}$.
QUESTION:	Is $\overrightarrow{\pi}$ Pareto-optimal?

Theorem 1. PO-ADD *is coNP-complete.*

Membership of coNP is easy to establish: a nondeterministic Turing machine could guess an allocation $\overrightarrow{\pi}'$, and check whether $\overrightarrow{\pi}'$ Pareto-dominates $\overrightarrow{\pi}$.

To prove coNP-hardness, we give a Karp reduction (*i.e.* polynomial time many-one reduction) from the coNP-complete language 3UNSAT.

Problem 2. Unsatisfiability of propositional 3CNF formulas (3UNSAT)	
INSTANCE:	A set of clauses C denoting a propositional formula in 3CNF.
QUESTION:	Is C unsatisfiable?

Let C be a set of propositional clauses of size 3 (we will suppose w.l.o.g. that the same literal does not appear more than once in each clause), $L(C)$ be the set of literals in C, and $V(C)$ be the set of variables in C. We will write $\mathcal{P}(C)$ to denote the following resource allocation instance:

Agents: $2|V(C)|+|C|+2$ agents: $\bigcup_{v\in V(C)}\{a_v, \overline{a_v}\}\cup\bigcup_{c\in C}\{a_c\}\cup\{a_{un}, a_{sat}\}$,

Objects: $4|C| + |V(C)| + 1$ objects: $\bigcup_{c\in C}\{o_{c,l} \mid l \in c\} \cup \bigcup_{v\in V(C)}\{o_v\} \cup \bigcup_{c\in C}\{o_c\} \cup \{o_{sat}\}$,

Preferences: $w(i, o) = 0$ for all i and all o, except:
- $w(a_v, o_v) = |\{c \mid v \in c \in C\}|$, and $w(\overline{a_v}, o_v) = |\{c \mid \neg v \in c \in C\}|$ for all $v \in V(C)$;
- $w(a_v, o_{c,v}) = 1$ if $v \in c$, and $w(\overline{a_v}, o_{c,\neg v}) = 1$ if $\neg v \in c$ for each $v \in V(C)$ and each $c \in C$;
- $w(a_c, o_{c,l}) = 1$ for each $c \in C$ and each $l \in c$;
- $w(a_c, o_c) = 1$ for all $c \in C$;
- $w(a_{un}, o_v) = 1$ for all $v \in V(C)$;
- $w(a_{sat}, o_c) = 1$ for all $c \in C$;
- $w(a_{un}, o_{sat}) = |V(C)| + 1$;
- $w(a_{sat}, o_{sat}) = |C|$.

Let I be a partial truth assignment of the variables in C. We will define its corresponding allocation $\overrightarrow{\pi}(I)$ as follows:

- $\pi(I)_{a_v} = \{o_v\}$ if $I(v) = \mathbf{true}$ and $\pi(I)_{a_v} = \{o_{c,v} \mid v \in c \in C\}$ otherwise, for each $v \in V(C)$;
- $\pi(I)_{\overline{a_v}} = \{o_v\}$ if $I(v) = \mathbf{false}$ and $\pi(I)_{\overline{a_v}} = \{o_{c,\neg v} \mid \neg v \in c \in C\}$ otherwise, for each $v \in V(C)$;
- for each $c \in C$: $\pi(I)_{a_c} = \{o_c\}$ if $I \not\models c$, and $\pi(I)_{a_c} = \{o_{c,l} \mid l \in C \wedge I \models l\}$ otherwise;

– $\pi(I)_{a_{un}} = \{o_{sat}\}$ if I is complete, and $\bigcup_{v \in V(C), I(v) \notin \{\mathbf{true}, \mathbf{false}\}} \{o_v\}$ otherwise;
– $\pi(I)_{a_{sat}} = \{o_{sat}\}$ if I is partial, and $\{o_c \mid I \vDash C\}$ otherwise.

It should be clear that for each assignment I, $\overrightarrow{\pi}(I)$ is well-defined.

Let $I_\emptyset$ be the empty assignment (*i.e.* the partial truth-assignment that leaves all variables unassigned). Our reduction transforms a 3UNSAT instance C into the PO-ADD-instance $\langle \mathcal{P}(C), I_\emptyset \rangle$.

We will now give an example of this reduction. Consider the 3UNSAT instance given by the set of clauses $\{c_1 = \{v_1, v_2, \neg v_3\}, c_2 = \{\neg v_1, \neg v_2, \neg v_3\}\}$.

If we run the reduction process on this instance, we get the PO-ADD-instance that is displayed in the table below. The columns of the table represent the agents and the rows of the table represent the items. The entries in the table are the weights. An entry is displayed in ***boldface italic*** and between brackets if the item of the corresponding row is allocated to the agent of the corresponding column. Empty cells in the table should be regarded as containing zero-weights.

	a_{c_1}	a_{c_2}	a_{v_1}	$\overline{a_{v_1}}$	a_{v_2}	$\overline{a_{v_2}}$	a_{v_3}	$\overline{a_{v_3}}$	a_{un}	a_{sat}
o_{v_1}			1	1					[***1***]	
o_{v_2}					1	1			[***1***]	
o_{v_3}								2	[***1***]	
o_{c_1}	[***1***]									1
o_{c_2}		[***1***]								1
o_{c_1,v_1}	1		[***1***]							
o_{c_1,v_2}	1				[***1***]					
$o_{c_1,\neg v_3}$	1							[***1***]		
$o_{c_2,\neg v_1}$		1		[***1***]						
$o_{c_2,\neg v_2}$		1				[***1***]				
$o_{c_2,\neg v_3}$		1						[***1***]		
o_{sat}									4	[***2***]

Lemma 1. *For each model M for C, $\overrightarrow{\pi}(M)$ Pareto-dominates $\overrightarrow{\pi}(I_\emptyset)$.*

Proof. In $\overrightarrow{\pi}(M)$, agent a_{un} has strictly higher utility: $u_{a_{un}}(\pi(M)_{a_{un}}) = |C| + 1$, while $u_{a_{un}}(\pi(I_\emptyset)_{a_{un}}) = |C|$. By definition of $\overrightarrow{\pi}(M)$, the utility of all other agents is in $\overrightarrow{\pi}(M)$ at least as high as in $\overrightarrow{\pi}(I_\emptyset)$. □

Lemma 2. *If C is unsatisfiable, then $\overrightarrow{\pi}(I_\emptyset)$ is Pareto-optimal.*

Proof. Suppose for contradiction that there is an allocation $\overrightarrow{\pi}'$ that Pareto-dominates $\overrightarrow{\pi}(I_\emptyset)$.

There is at least one agent a such that $u_a(\pi'_a) > u_a(\pi(I_\emptyset)_a)$. It can be easily proved that, starting from $I_\emptyset$, strictly increasing the utility of any agent in $A \setminus \{a_{un}, a_{sat}\}$ implies reallocating at least one item o_v from agent a_{un}'s share to another agent. Then, the only solution for not decreasing a_{un}'s utility is to give her o_{sat}.

So then $\{o_c \mid c \in C\} \in \pi'_{a_{sat}}$. Consequently, for all a_c with $c \in C$ we must have $|\{o_{c,l} \mid l \in c \wedge o_{c,l} \in \pi'_{a_c}\}| \geq 1$. Let $o \in \{o_{c,l} \mid l \in c \wedge o_{c,l} \in \pi'_{a_c}\}$. Let a' be the agent for which it holds that $o \in \pi(I_\emptyset)_{a'}$ (so $a' \in \{a_v, \overline{a_v}\}$). Let $\overline{a'}$ be the agent in $\{a_v, \overline{a_v}\}$ that does not equal a'. It must now be that $o_v \in \pi'_{a'}$, and as a consequence we now know that $\pi(I_\emptyset)_{\overline{a}} = \pi'_{\overline{a}}$.

So, for each $v \in V(C)$ there are no two objects $o_{c,v}$ and $o_{c',\neg v}$ with $v \in c$, $\neg v \in c'$, and $c, c' \in C$ such that $o_{c,v} \in \pi_{a_c}$ and $o_{c',\neg v} \in \pi'_{a_{c'}}$. It then follows immediately from the construction of the reduction that there is a complete interpretation I such that $\overrightarrow{\pi}' = \overrightarrow{\pi}(I)$. Moreover, one can check that $I \vDash c$ for each $c \in C$. Therefore, I is a model of C, and we have a contradiction. □

Proof (Theorem 1). Let C be a set of clauses of size 3. By Lemma 1 and 2, we have that C is unsatisfiable if and only if $\overrightarrow{\pi}(I_\emptyset)$ is a Pareto-optimal allocation for $\mathcal{P}(C)$. The reduction from C to $\langle \mathcal{P}(C), \overrightarrow{\pi}(I_\emptyset)\rangle$ can clearly be done in polynomial time, hence coNP-hardness is proved. □

4 Complexity of Deciding Existence of Efficient and Envy-Free Allocations for Agents with Additive Utility

For this section, we are interested in finding allocations that are both Pareto-efficient and envy-free.

We will now state the problem and prove that this problem is Σ_2^p-complete.

Problem 3. Efficient & envy-free allocation existence with additive utility functions (∃-EEF-ADD)

INSTANCE:	A resource allocation instance $\mathcal{P} = \langle A, O, w\rangle$.
QUESTION:	Does there exist an allocation that is both Pareto-efficient and envy-free?

Theorem 2. ∃-EEF-ADD *is* Σ_2^p*-complete.*

We will prove Σ_2^p-completeness by a Karp reduction from the complement of the Π_2^p-complete language ∀∃3CNF [12].

Problem 4. Doubly quantified 3CNF satisfiability (∀∃3CNF)

INSTANCE:	A set $V_\forall$ of propositional variables, a set $V_\exists$ of propositional variables, a set C of clauses of three literals over the variables $V_\forall \cup V_\exists$.
QUESTION:	Does there exist for each assignment to the variables in $V_\forall$ an assignment to the variables in $V_\exists$ that satisfies C?

Let $F = \langle V_\forall, V_\exists, C\rangle$ be an instance of ∀∃3CNF. We will assume w.l.o.g. that every possible literal occurs at least once in C, and that a literal does not appear more than once in each clause. Let #occ∀ be the number of literal occurrences in C of variables in $V_\forall$. We will write $L_\forall$ and $L_\exists$ for the sets of literals of variables in $V_\forall$ and $V_\exists$ respectively. We will write $\mathcal{P}(F)$ to denote the following resource allocation instance (also see the example that follows after Definition 6):

Agents: $4|V_\forall| + 2|V_\exists| + |C| + \#\text{occ}\forall + 3$ agents: $\bigcup_{v \in V_\forall}\{a_v^+, a_v^{+cl}, a_v, \overline{a_v}\} \cup \bigcup_{v \in V_\exists}\{a_v, \overline{a_v}\} \bigcup_{c \in C}\{a_c\} \cup \bigcup_{c \in C}\{a_{c,l}^{en} \mid l \in c \cap L_\forall\} \cup \{a_{un}, a_{sat}, a_{en}\}$

Objects: $4|V_\forall| + |V_\exists| + 5|C| + L_\forall + 3$ objects: $\bigcup_{v \in V_\forall}\{o_v^{en}, o_v^{cmp}, o_v^h, \overline{o_v^h}\} \cup \bigcup_{v \in V_\exists}\{o_v\} \cup \bigcup_{c \in C, l \in c}\{o_c, o_c^{cmp} o_{c,l}\} \cup \bigcup_{c \in C}\{o_{c,l}^{en} \mid l \in c \cap L_\forall\} \cup \{o_{sat}, o_{en1}, o_{en2}\}$

Preferences: $w(i, o) = 0$ for all i and all o, except:

- For all $v \in V_\forall$ and $o \in \{o_v^{en}, o_v^c, o_v^h, \overline{o_v^h}\}$: $w(a_v^+, o) = 1$ and $w(a_v^{+cl}, o) = 1$;[3]
- For all $v \in V_\forall$: $w(a_v, o_v^h) = |\{c \mid v \in c\}|$, $w(\overline{a_v}, \overline{o_v^h}) = |\{c \mid \neg v \in c\}|$, $w(a_v, o_{c,v}) = 1$ for all $c \in C$ where $v \in c$, and $w(\overline{a_v}, o_{c,\neg v}) = 1$ for all $c \in C$ where $\neg v \in c$;
- For all $v \in V_\exists$: $w(a_v, o_v) = |\{c \mid v \in c\}|$, $w(\overline{a_v}, o_v) = |\{c \mid \neg v \in c\}|$, $w(a_v, o_{c,v}) = 1$ for all $c \in C$ where $v \in c$, and $w(\overline{a_v}, o_{c,\neg v}) = 1$ for all $c \in C$ where $\neg v \in c$;
- For all $c \in C$: $w(a_c, o_c) = M$, $w(a_c, o_c^{cmp}) = M - 1$, and $w(a_c, o_{c,l}) = 1$ for all $l \in c$;
- For all (c, l) where $c \in C$, $l \in c \cap L_\forall$: $w(a_{c,l}^{en}, o_c) = M$, $w(a_{c,l}^{en}, o_{c,l}) = 1$, $w(a_{c,l}^{en}, o_{c,l}^{en}) = M$;
- For all $c \in C$: $w(a_{sat}, o_c) = 1$, $w(a_{sat}, o_{sat}) = |C|$, $w(a_{sat}, o_{en1}) = \frac{1}{2}$;
- For all $c \in C$: $w(a_{un}, o_c^{cmp}) = 1$; For all $v \in V_\forall$: $w(a_{un}, o_v^{cmp}) = 1$; For all $v \in V_\exists$: $w(a_{un}, o_v) = 1$;
- $w(a_{un}, o_c^{sat}) = |V_\exists| + |V_\forall| + |C| + 1$; $w(a_{un}, o_{en1}) = 2(|V_\exists| + |V_\forall| + |C| + 1)$; $w(a_{un}, o_{en2}) = 3(|V_\exists| + |V_\forall| + |C| + 1) - 1$; $w(a_{en}, o_{en2}) = M$;

where M is a large number. It suffices to take for M the sum of all weights that are not defined in terms of M.

For our proof that $\exists$-EEF-ADD is Σ_2^p-complete, we need the notion of a special type of allocation for $\mathcal{P}(F)$. An example of an $X_\forall$-allocation is given in the example-instance that follows after this proof.

Definition 6 ($V_\forall$-assignments and $V_\forall$-allocations). *For F, we define a $V_\forall$-assignment as any complete assignment to the variables in $V_\forall$ only. Given a $V_\forall$-assignment I, we define a corresponding allocation $\overrightarrow{\pi}(I)$ for $\mathcal{P}(F)$ in the following way:*

- $\pi(I)_{a_v^+} = \{o_v^{en}\}$ *for each* $v \in V_\forall$;
- $\pi(I)_{a_v^{+cl}} = \{o_v^h\}$ *if* $I(v) =$ **true** *and* $\{\overline{o_v^h}\}$ *otherwise, for each* $v \in V_\forall$;
- $\pi(I)_{a_v} = \{o_{c,v} \mid v \in c\} \cup \{o_v^h\}$ *if* $I(v) =$ **false**, *and* $\{o_{c,v} \mid v \in c\}$ *otherwise, for each* $v \in V_\forall$;
- $\pi(I)_{\overline{a_v}} = \{o_{c,\neg v} \mid \neg v \in c\} \cup \{\overline{o_v^h}\}$ *if* $I(v) =$ **true**, *and* $\{o_{c,\neg v} \mid \neg v \in c\}$ *otherwise, for each* $v \in V_\forall$;
- $\pi(I)_{a_v} = \{o_{c,l} \mid l \in c\}$ *for each* $l \in L_\exists$;
- $\pi(I)_{a_c} = \{o_c\}$ *for each* $c \in C$;
- $\pi(I)_{a_{c,l}^{en}} = \{o_{c,l}^{en}\}$ *for each* c, l *where* $c \in C$ *and* $l \in c \cap L_\forall$;
- $\pi(I)_{a_{un}} = \{o_c^{cmp} \mid c \in C\} \cup \{o_v \mid v \in V_\exists\} \cup \{o_v^{cmp} \mid v \in V_\forall\} \cup \{o_{en1}\}$;

[3] So, a_v^{+cl} is a *clone* of a_v^+.

– $\pi(I)_{a_{sat}} = \{o_{sat}\}$;
– $\pi(I)_{a_{en}} = \{o_{en2}\}$;

Given a $V_\forall$-assignment I, we define the set of $V_\forall$-allocations corresponding to I as follows: any allocation that can be obtained from $\pi(I)$ by a sequence of swaps of the bundles of a_v^+ and a_v^{+cl} for any $v \in V_\forall$, followed by a sequence of reallocations of $o_{c,l}$ to $a_{c,l}^{en}$ for any l, c with $c \in C$, $l \in c \cap L_\forall$, and $I \not\models l$.[4]

Let us give an example of this reduction, together with a $V_\forall$-allocation. Let the $\forall\exists$3CNF-instance be $F = \langle V_\forall = \{v_1\}, V_\exists = \{v_2\}, C = \{c_1 = \{v_1, \neg v_1, v_2\}, c_2 = \{v_2, \neg v_2, v_1\}\}\rangle$. Then $\mathcal{P}(F)$ looks as follows.

	a_{c_1}	a_{c_2}	$a_{v_1}^+$	$a_{v_1}^{+cl}$	a_{v_1}	$\overline{a_{v_1}}$	a_{v_2}	$\overline{a_{v_2}}$	a_{c_1,v_1}^{en}	$a_{c_1,\neg v_1}^{en}$	a_{c_2,v_1}^{en}	a_{un}	a_{sat}	a_{en}
o_{c_1}	***[M]***								M	M			1	
o_{c_2}		***[M]***									M		1	
$o_{c_1}^{cmp}$	M-1											***[1]***		
$o_{c_2}^{cmp}$		M-1										***[1]***		
o_{c_1,v_1}	1				1				***[1]***					
$o_{c_1,\neg v_1}$	1					***[1]***				1				
o_{c_1,v_2}	1						***[1]***							
o_{c_2,v_2}		1					***[1]***							
$o_{c_2,\neg v_2}$		1						***[1]***						
o_{c_2,v_1}		1			1						***[1]***			
$o_{v_1}^h$			1	1	***[2]***									
$\overline{o_{v_1}^h}$			***[1]***	1		1								
$o_{v_1}^{en}$			1	***[1]***										
$o_{v_1}^{cmp}$			1	1								***[1]***		
o_{v_2}							2	1				***[1]***		
o_{c_1,v_1}^{en}									***[M]***					
$o_{c_1,\neg v_1}^{en}$										***[M]***				
o_{c_2,v_1}^{en}											***[M]***			
o_{sat}												5	***[2]***	
o_{en1}												***[10]***	$\frac{1}{2}$	
o_{en2}												14		***[M]***

A $V_\forall$-allocation corresponding to a $V_\forall$-assignment I with $I(v_1) =$ **false** is displayed in ***boldface italic*** and between brackets. This allocation has been obtained from $\pi(I)$ by swapping the bundles of $a_{v_1}^+$ and $a_{v_1}^{+cl}$, and reallocating item o_{c_1,v_1} to a_{c_1,v_1}^{en}, and item o_{c_2,v_1} to a_{c_2,v_1}^{en}.

In the following proofs, we will restrict attention to *non-wasting* allocations, that is, allocations π such that for all $(o, a) \in O \times A$, $o \in \pi(a) \Rightarrow w(a, o) > 0$. It is obvious that every Pareto-efficient allocation is a non-wasting one.

Lemma 3. *Let $\overrightarrow{\pi}$ be an allocation. $\overrightarrow{\pi}$ is envy-free if and only if $\overrightarrow{\pi}$ is a $V_\forall$-allocation.*

Proof. ($\Leftarrow$) For any arbitrary $V_\forall$-allocation, it is easy (although a bit tedious) to check for each type of agent that she does not envy any other agent.

($\Rightarrow$) We show this by reasoning about how resources should be allocated in order to prevent envy. We start by noticing that o_{en2} must necessarily be allocated to a_{en}. As a consequence o_{en1} must be allocated to a_{un}, after which

[4] To remove any confusion, see the example allocation that follows, together with the explanation.

o_{sat} should go to a_{sat}. In order prevent a_{un} from envying a_{en}, we should give a_{un} all remaining items for which his weight is positive. Now that we know we cannot allocate o_c^{cmp} to a_c (for any $c \in C$), we must give o_c to a_c. For the same reason, for all $v \in V_\exists$ we cannot give o_v to a_v or $\overline{a_v}$, thus these agents should receive all remaining resources for which they have a positive weight. Next, we notice that we should give $o_{c,l}^{en}$ to $a_{c,l}^{en}$ because $a_{c,l}^{en}$ is the only agent with positive weight for this item. Now we see that for all c, l with $l \in c \cap L_\forall$ and $c \in C$, we cannot give $o_{c,l}$ to a_c, because $a_{c,l}^{en}$ would then envy a_c. Next, for all $v \in V_\forall$ we must allocate o_v^{en} to either a_v^+ or a_v^{+cl}. Because the weights of both agents are exactly the same, suppose w.l.o.g. that we allocate o_v^{en} to a_v^+. In order to prevent envy between the two agents, we need to allocate either o_v^h or $\overline{o_v^h}$ (but not both) to a_v^{+cl} (we denote by o this item and $\overline{o}$ the other one of the pair). Since $\overline{o}$ cannot be allocated to a_v^+ nor to a_v^{+cl}, $\overline{o}$ must go to a_v if $\overline{o} = o_v^h$ and $\overline{a_v}$ otherwise. There is only one agent left that we can allocate o to. Lastly, let $a \in \{a_v, \overline{a_v}\}$ be the agent that does not get $\overline{o}$. All items that a has positive weight for should now be allocated to a, in order to prevent a from envying the agent that gets $\overline{o}$.

The restrictions that we just deduced, restrict the set of possibly efficient and envy-free allocations to the set of $V_\forall$-allocations. □

Lemma 4. *No two $V_\forall$-allocations dominate each other.*

Proof. Let $\overrightarrow{\pi}$ and $\overrightarrow{\pi}'$ be two $V_\forall$-allocations. If $\overrightarrow{\pi}$ and $\overrightarrow{\pi}'$ correspond to the same $V_\forall$-assignment, then $\overrightarrow{\pi}$ does not dominate $\overrightarrow{\pi}'$ because swapping the bundles of o_v^+ and o_v^{+cl} for any $v \in V_\forall$ does not increase nor decrease the utility of both agents. Reallocating $o_{c,l}$ between the agents $a_{c,l}^{en}$ and a_v (or $\overline{a_v}$) for any l, c with $l \in c \cap L_\forall$ and $c \in C$ can never result in a dominating allocation either, because both agents have exactly the same weights for all of these items.

For the case that $\overrightarrow{\pi}$ and $\overrightarrow{\pi}'$ correspond to different $V_\forall$-assignments, let I and I' be the two $V_\forall$-assignments respectively, and let v be a variable such that $I(v) \neq I'(v)$. We will show that $\overrightarrow{\pi}$ does not Pareto-dominate $\overrightarrow{\pi}'$. Assume w.l.o.g. that $I(v) = \textbf{true}$ and $I'(v) = \textbf{false}$. In $\overrightarrow{\pi}$ we have that $\overline{o_v^h}$ is allocated to $\overline{a_v}$; in $\overrightarrow{\pi}'$ this is not the case. Because of the weights that $\overline{a_v}$ has, we now know that $u_{\overline{a_v}}(\pi_{\overline{a_v}}) \leq u_{\overline{a_v}}(\pi'_{\overline{a_v}})$. We can divide this up in two cases: in the case that $u_{\overline{a_v}}(\pi_{\overline{a_v}}) < u_{\overline{a_v}}(\pi'_{\overline{a_v}})$, we have immediately that $\overrightarrow{\pi}$ does not Pareto-dominate $\overrightarrow{\pi}'$. In the other case that $u_{\overline{a_v}}(\pi_{\overline{a_v}}) = u_{\overline{a_v}}(\pi'_{\overline{a_v}})$, any item $o_{c,\neg v}$ in the set $\{o_{c,\neg v} \mid v \in c \in C\}$ is allocated to $a_{c,\neg v}^{en}$ under allocation $\overrightarrow{\pi}$, but not under allocation $\overrightarrow{\pi}'$, so in this case we have $u_{\overline{a_{c,\neg v}}}(\pi_{\overline{a_{c,\neg v}}}) < u_{\overline{a_{c,\neg v}}}(\pi'_{\overline{a_{c,\neg v}}})$, hence $\overrightarrow{\pi}$ does not Pareto-dominate $\overrightarrow{\pi}'$. □

Lemma 5. *Given a $V_\forall$-assignment I for F, and a $V_\forall$-allocation $\overrightarrow{\pi}$ for $\mathcal{P}(F)$ that corresponds to I; if C is satisfiable on I (i.e. I can be extended such that C is satisfied), then there is an allocation $\overrightarrow{\pi}'$ that Pareto-dominates $\overrightarrow{\pi}$.*

Proof. Let I' be a complete assignment that satisfies C such that $I \subseteq I'$. The following allocation $\overrightarrow{\pi}'$ Pareto-dominates $\overrightarrow{\pi}$.

– For all $c \in C$: $\pi'_c = \{o_{c,l} \mid l \in c \wedge I' \vDash l\} \cup \{o_c^{cmp}\}$.
– For all $v \in V_\forall$: Let $a = a_v$, $\overline{a} = \overline{a_v}$, $o = o_v^h$ if $I(v) = \textbf{true}$, and let $a = \overline{a_v}$, $\overline{a} = a_v$, $o = \overline{o_v^h}$ otherwise. $\pi'_a = \{o\}$ and $\pi'_{\overline{a}} = \pi_{\overline{a}}$. Moreover, if $o \in \pi_{a_v^+}$ then $\pi'_{a_v^+} = \pi_{a_v^+} \backslash \{o\} \cup \{o_v^{cmp}\}$ and $\pi'_{a_v^{+cl}} = \pi_{a_v^{+cl}}$; otherwise $\pi'_{a_v^{+cl}} = \pi_{a_v^{+cl}} \backslash \{o\} \cup \{o_v^{cmp}\}$ and $\pi'_{a_v^+} = \pi_{a_v^+}$.
– For all $v \in V_\exists$: If $I'(v) = \textbf{true}$, then $\pi'_{a_v} = \{o_v\}$ and $\pi'_{\overline{a_v}} = \pi_{\overline{a_v}}$; otherwise $\pi'_{\overline{a_v}} = \{o_v\}$ and $\pi'_{a_v} = \pi_{a_v}$.
– For all c, l such that $c \in C$ and $l \in c \cap L_\forall$: $\pi'_{a_{c,l}^{en}} = \pi_{a_{c,l}^{en}}$.
– $\pi'_{a_{un}} = \{o_{sat}, o_{en1}\}$; $\pi'_{a_{sat}} = \{o_c \mid c \in C\}$; $\pi'_{a_{en}} = \{o_{en2}\}$.

In $\overrightarrow{\pi}'$, the utility of a_{un} is strictly higher than in $\overrightarrow{\pi}$. Moreover, one can easily check that in $\overrightarrow{\pi}'$ the utilities of all other agents are at least as high as in $\overrightarrow{\pi}$. □

Lemma 6. *Given a $V_\forall$-assignment I for F, and a $V_\forall$-allocation $\overrightarrow{\pi}$ in $\mathcal{P}(F)$ that corresponds to I; if C is unsatisfiable on I, then $\overrightarrow{\pi}$ is Pareto-efficient.*

Proof. We will first show that in any $\overrightarrow{\pi}'$ that Pareto-dominates $\overrightarrow{\pi}$ we necessarily have $\{o_c \mid c \in C\} \subseteq \pi'_{a_{sat}}$. We do this by exhaustion on the type of agent. Let $\overrightarrow{\pi}'$ Pareto-dominate $\overrightarrow{\pi}$. Let a be an agent such that $u_a(\pi'_a) > u_a(\pi_a)$. We show for each of the following cases of a that necessarily $\{o_c \mid c \in C\} \subseteq \pi'_{a_{sat}}$.

Case 1: $a = a_{un}$**:** In this case we clearly have $o_{sat} \in \pi'_a$, and because $w(a_{sat}, o_{sat}) = |C|$ we have as a consequence that $\{o_c \mid c \in C\} \subseteq \pi'_{a_{sat}}$.
Case 2: $a \in \{a_v, \overline{a_v} \mid v \in V_\exists\}$**:** Let $a = a_v$ for an arbitrary $v \in V_\exists$ (the case that $a = \overline{a_v}$ is analogous). It must be that $o_v \in \pi'_a$, so then, since o_v has been removed from a_{un}'s share, we need to give her o_{sat} as a compensation. From the argument in the previous case we get $\{o_c \mid c \in C\} \subseteq \pi'_{a_{sat}}$.
Case 3: $a \in \{a_v^+ \, a_v^{+cl} \mid v \in V_\forall\}$**:** Let $a = a_v^+$ for an arbitrary $v \in V_\forall$ (the case that $a = a_v^{+cl}$ is analogous). Assume w.l.o.g. that $I(v) = \textbf{true}$. Because a_v^+ and a_v^{+cl} have identical weights, we may also w.l.o.g. assume that $\pi'_{a_v^+} \cap \pi_{a_v^{+cl}} = \emptyset$. If $o_v^{cmp} \in \pi'_a$, then from the argument in case 1 it follows that $\{o_c \mid c \in C\} \subseteq \pi'_{a_{sat}}$. If $\overline{o_v^h} \in \pi'_a$, then it must be that $\{o_{c,\neg v} \mid \neg v \in c \in C\} \cap \pi_{\overline{a_v}} = \emptyset$ and $\{o_{c,\neg v} \mid \neg v \in c \in C\} \subseteq \pi'_{\overline{a_v}}$. Thus, for all $c \in C$ with $\neg v \in c$, we have $o_c \in \pi'_{a_{c,\neg v}^{en}}$. Consequently, we get $o_c^{cmp} \in \pi'_{a_c}$; and therefore by our argument that we gave in Case 1, $\{o_c \mid c \in C\} \subseteq \pi'_{a_{sat}}$.
Case 4: $a \in \{a_v, \overline{a_v} \mid v \in V_\forall\}$**:** Assume w.l.o.g. that $I(v) = \textbf{true}$. For any arbitrary $v \in V_\forall$, let $a = \overline{a_v}$. From the last part of the argument that we gave for the previous case, it follows directly that $\{o_c \mid c \in C\} \subseteq \pi'_{a_{sat}}$. Now let $a = a_v$. Necessarily we have $o_v^h \in \pi'_{a_v}$, and from our reasoning in Case 3 it follows that $\{o_c \mid c \in C\} \subseteq \pi'_{a_{sat}}$.
Case 5: $a \in \{a_c \mid c \in C\}$**:** Let $a = a_c$ for an arbitrary $c \in C$. If $o_c^{cmp} \in \pi'_{a_c}$, it follows from Case 1 that $\{o_c \mid c \in C\} \subseteq \pi'_{a_{sat}}$. If $o_{c,l} \in \pi'_{a_c}$ then the same conclusion follows, but this time from the last part of the proof of Case 3.
Case 6: $a \in a_{c,l}^{en} \mid c \in C \wedge l \in c \cap L_\forall\}$**:** Let $a = a_{c,l}^{en}$ for an arbitrary c, l with $c \in C$ and $l \in c \cap L_\forall$. We must have that $o_c \in \pi'_{a_{c,l}^{en}}$ or $o_{c,l} \in \pi'_{a_{c,l}^{en}}$ (or both), in both cases it follows from the last part of the proof of Case 3 that $\{o_c \mid c \in C\} \subseteq \pi'_{a_{sat}}$.

Case 7: $a = a_{sat}$**:** Let C' be any strict subset of C. If $\{o_c \mid c \in C'\} \subseteq \pi'_a$, then $\forall c \in C : o_c^{cmp} \in a_c$, so by the arguments in Case 1, $\{o_c \mid c \in C\} \subseteq \pi'_a$. If $o_{en1} \in \pi'_a$, then the same follows, also from the proof in Case 1.

Case 8: $a = a_{en}$**:** This case is obviously impossible.

Now we will finish the proof by obtaining the contradiction that an extension I' of I can be made to the variables in $V_\exists$, such that I' satisfies C.

Recall that we assume that $\overrightarrow{\pi}'$ Pareto-dominates $\overrightarrow{\pi}$, and as we have just shown, $\{o_c \mid c \in C\} \subseteq \pi'_{a_{sat}}$. For all $c \in C : o_c^{cmp} \cup L_c \subseteq \pi'_{a_c}$, where L_c is any subset of $\{o_{c,l} \mid l \in c\}$. Let $o_{c,l} \in L_c$ and let $v \in l$. There are two cases: either $v \in V_\forall$ or $v \in V_\exists$.

Suppose $v \in V_\forall$. Let $a = a_v$, $\overline{a} = \overline{a_v}$ if $I(v) = \mathbf{true}$, and let $a = \overline{a_v}$, $\overline{a} = a_v$ otherwise. It is easy to see that $\nexists o_{c,l} \in L_c : o_{c,l} \in \pi_{a^{en}_{c,l}}$ and $\nexists o_{c,l} \in L_c : o_{c,l} \in \pi_{\overline{a}}$, so $o_{c,l} \in \pi_a$. As a consequence, we know that l is satisfied by I. Hence, it must hold that if $o_{c,l} \in L_c$, then c is satisfied by I.

Suppose $v \in V_\exists$. Let $a = \overline{a_v}$ if $l = \neg v$ and let $a = a_v$ otherwise. Then it must be that $o_{c,l} \in \pi_a$ and $o_v \in \pi'_a$.

From the construction of the reduction, it follows that there must exist an assignment to the variables in $V_\exists$ that satisfies all clauses not satisfied by I, *i.e.*, we obtain the contradiction that C is satisfiable on I. □

Proof (Theorem 2). Membership is easily established: A nondeterministic $\mathsf{NP}^{\mathsf{NP}}$ Turing machine that decides this problem could work as follows. On input $\langle A, \mathcal{O}, w\rangle$:

1. Guess an allocation $\overrightarrow{\pi}$.
2. Check whether $\overrightarrow{\pi}$ is envy-free. If not, then REJECT.
3. Check whether $\overrightarrow{\pi}$ is Pareto-optimal by querying the oracle. If it is, then ACCEPT. Otherwise, REJECT.

The difficult part is proving Σ_2^p-hardness.

Given a $\forall\exists$3CNF-instance $F = \langle V_\forall, V_\exists, C\rangle$, we can clearly construct $\mathcal{P}(F)$ in polynomial time.

If F is a NO-instance of $\forall\exists$3CNF, then there is a $V_\forall$-assignment I that cannot be extended to an assignment that satisfies C. Let $\overrightarrow{\pi}$ be a $V_\forall$-allocation for $\mathcal{P}(F)$ that corresponds to I. By Lemma 3, $\overrightarrow{\pi}$ is envy-free and by Lemma 6, $\overrightarrow{\pi}$ is Pareto-efficient. Hence, $\mathcal{P}(F)$ is a YES-instance of $\exists$-EEF-ADD.

If F is a YES-instance of $\forall\exists$3CNF, then for any $V_\forall$-assignment I that we pick, C is satisfiable on I. Let $\overrightarrow{\pi}$ be any $V_\forall$-allocation for $\mathcal{P}(F)$ that corresponds to I. By Lemma 5, there is an allocation $\overrightarrow{\pi}'$ that Pareto-dominates $\overrightarrow{\pi}$. By Lemma 4, $\overrightarrow{\pi}$ is not a $V_\forall$-allocation. Finally, because $\overrightarrow{\pi}$ is not a $V_\forall$-allocation, it follows from Lemma 3 that $\overrightarrow{\pi}'$ is not envy free. Because we had taken $\overrightarrow{\pi}$ to be an arbitrary $V_\forall$-allocation for an arbitrary $V_\forall$-assignment I, it follows that $\mathcal{P}(F)$ is a NO-instance of $\exists$-EEF-ADD.

Therefore we conclude that $\exists$-EEF-ADD is Σ_2^p-hard. □

5 Discussion

We have introduced in this paper two new complexity results for the resource allocation problem with additive preferences, thus filling an important gap in the previous complexity studies of this problem, mainly in [8]. Our main result shows that, even with very simple preferences (additive), deciding whether there is a Pareto-efficient and envy-free allocation is computationally very hard. This goes slightly beyond the results in [8], as it shows that the high complexity of the problem is not only related to the presence of preferential dependencies (complementarity or substitutability) between objects, since the hardness holds under the assumption of additive independence.

There are several natural ways of overcoming this high complexity. The first one could be to impose some restrictions on the setting to decrease the complexity. However, as stated in [8], the natural restrictions of the problem imply a huge loss of generality, and thus are of limited practical interest. Another solution is to relax envy-freeness or Pareto-efficiency, such as in [7], where envy-freeness is replaced by a measure of envy, and where allocations are only required to be *complete* (that is, all objects must be allocated) instead of being Pareto-efficient. An idea could be to mix collective utility maximization (*e.g.* classical utilitarian or egalitarian) with envy-minimization.[5] And lastly, designing efficient approximation algorithms could be a way of getting around the high complexity of the problem.

References

1. Chevaleyre, Y., Dunne, P.E., Endriss, U., Lang, J., Lemaître, M., Maudet, N., Padget, J., Phelps, S., Rodríguez-Aguilar, J.A., Sousa, P.: Issues in multiagent resource allocation. Informatica 30, 3–31 (2006); Survey paper
2. Brams, S.J., Taylor, A.: Fair Division: From Cake-Cutting to Dispute Resolution. Cambridge Univ. Press, Cambridge (1996)
3. Demko, S., Hill, T.P.: Equitable distribution of indivisible items. Mathematical Social Sciences 16, 145–158 (1998)
4. Shehory, O., Kraus, S.: Methods for Task Allocation via Agent Coalition Formation. Artificial Intelligence 101(1-2), 165–200 (1998)
5. Modi, P.J., Jung, H., Tambe, M., Shen, W.M., Kulkarni, S.: A dynamic distributed constraint satisfaction approach to resource allocation. In: Walsh, T. (ed.) CP 2001. LNCS, vol. 2239, pp. 685–700. Springer, Heidelberg (2001)
6. de Weerdt, M.M., Zhang, Y., Klos, T.B.: Distributed task allocation in social networks. In: Huhns, M., Shehory, O. (eds.) Proceedings of the 6th International Conference on Autonomous Agents and Multiagent Systems, Bradford, UK, pp. 488–495. IFAAMAS, Research Publishing Services (2007)
7. Lipton, R.J., Markakis, E., Mossel, E., Saberi, A.: On approximately fair allocations of indivisible goods. In: EC 2004: Proceedings of the 5th ACM conference on Electronic commerce, pp. 125–131. ACM Press, New York (2004)

[5] A similar idea is proposed in [13].

8. Bouveret, S., Lang, J.: Efficiency and envy-freeness in fair division of indivisible goods: Logical representation and complexity. Journal of Artificial Intelligence Research (JAIR) 32, 525–564 (2008)
9. Dunne, P.E., Wooldridge, M., Laurence, M.: The complexity of contract negotiation. Artificial Intelligence 164(1-2), 23–46 (2005)
10. Chevaleyre, Y., Endriss, U., Estivie, S., Maudet, N.: Multiagent resource allocation with k-additive utility functions. In: Proceedings of the First International Workshop on Computer Science and Decision Theory, Paris, France, pp. 83–100 (2004)
11. Conitzer, V., Sandholm, T., Santi, P.: Combinatorial auctions with k-wise dependent valuations. In: Proceedings of the 20th National Conference on Artificial Intelligence (AAAI 2005), pp. 248–254. AAAI Press, Menlo Park (2005)
12. Schaefer, M., Umans, C.: Completeness in the polynomial-time hierarchy: a compendium. SIGACT News (September 2002)
13. Brams, S.J., King, D.L.: Efficient fair division: Help the worst off or avoid envy? Rationality and Society 17(4), 387–421 (2005)