

Multivariate Algorithmics for Eliminating Envy by Donating Goods

Niclas Boehmer
TU Berlin
Berlin, Germany
niclas.boehmer@tu-berlin.de

Robert Brederbeck
Humboldt-Universität zu Berlin
Berlin, Germany
robert.brederbeck@hu-berlin.de

Klaus Heeger
TU Berlin
Berlin, Germany
heeger@tu-berlin.de

Dušan Knop
Czech Technical University in Prague
Prague, Czech Republic
dusan.knop@fit.cvut.cz

Junjie Luo
Nanyang Technological University
Singapore
junjie.luo@ntu.edu.sg

ABSTRACT

Fairly dividing a set of indivisible resources to a set of agents is of utmost importance in some applications. However, after an allocation has been implemented the preferences of agents might change and envy might arise. We study the following problem to cope with such situations: Given an allocation of indivisible resources to agents with additive utility-based preferences, is it possible to socially donate some of the resources (which means removing these resources from the allocation instance) such that the resulting modified allocation is envy-free (up to one good). We require that the number of deleted resources and/or the caused utilitarian welfare loss of the allocation are bounded. We conduct a thorough study of the (parameterized) computational complexity of this problem considering various natural and problem-specific parameters (e.g., the number of agents, the number of deleted resources, or the maximum number of resources assigned to an agent in the initial allocation) and different preference models, including unary and 0/1-valuations. In our studies, we obtain a rich set of (parameterized) tractability and intractability results and discover several surprising contrasts, for instance, between the two closely related fairness concepts envy-freeness and envy-freeness up to one good and between the influence of the parameters maximum number and welfare of the deleted resources.

KEYWORDS

Fair Allocation; Indivisible Resources; Envy-Freeness; Donating Goods; Parameterized Algorithmics

ACM Reference Format:

Niclas Boehmer, Robert Brederbeck, Klaus Heeger, Dušan Knop, and Junjie Luo. 2022. Multivariate Algorithmics for Eliminating Envy by Donating Goods. In *Proc. of the 21st International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2022), Online, May 9–13, 2022*, IFAAMAS, 9 pages.

1 INTRODUCTION

Zarah is in big troubles due to numerous complains about an unfair allocation of resources. Alice thinks that Bob is much better off because of his new screen and laptop. Bob and Carol explain that

Dan’s room is way bigger than that of everyone else and that only his key can unlock the backdoor. Dan complains about Alice owning a new tablet and keyboard, while everyone admires Carol’s mouse and that she has a large fridge in her room just for herself. While we leave it to the reader whether the protesters are Zarah’s children, PhD students or employees, we remark that this situation needs to be cleared quickly, because envy blocks Zarah’s protégés from doing anything other than complaining. Since an envy-free reallocation of the resources turned out to be impossible in her case, Zarah implements another effective solution: She decides that Carol’s fridge is now usable by everyone and puts it into Dan’s room to take away his extra space while donating Alice’s keyboard and Bob’s screen to the orphanage. Doing so, Zarah completely eliminates all envy between her protégés.

Real-world allocations, as in our toy example, are often not envy-free for various reasons (even in envy-free allocations, envy can emerge if preferences change over time). While reallocating resources might generally be an option, this can be very expensive or even impossible (e.g., Alice’s room might be too small for Carol’s fridge). Moreover, envy-free (re)allocations may simply not exist. Nevertheless, the need for envy-freeness is undoubtful in some applications (such as heritage or divorce disputes), so that every possible way out should be considered. This work focuses on one of the most natural such possibilities: Given an allocation of resources, we ask to make the allocation envy-free (EF) or envy-free up to one good (EF1) by donating, that is, taking away some of the resources. Since empty allocations are envy-free but obviously unwanted, we additionally aim for an upper bound on the number of resources being donated and/or on the social welfare decrease.

There is also a more subtle interpretation of donating a resource as sharing it, i.e., making it accessible to everyone. If reallocating resources is impossible or not sufficient, then making a resource accessible to everyone (like Carol’s fridge in our example) can be a very attractive way out, increasing the overall social welfare.

1.1 Related Work

We are not aware of previous work capturing exactly our setting but remark that the idea of improving the fairness of an allocation is trending and considered in many different ways and models. We give an overview of the most related work.

Segal-Halevi [23] studied a model similar to ours for divisible resources, where an initial *unfair* allocation of a cake is given, and

Proc. of the 21st International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2022), P. Faliszewski, V. Mascardi, C. Pelachaud, M.E. Taylor (eds.), May 9–13, 2022, Online. © 2022 International Foundation for Autonomous Agents and Multiagent Systems (www.ifaamas.org). All rights reserved.

the goal is to redivide the cake to balance fairness and ownership rights. In contrast to our work, he considered divisible resources (in contrast to indivisible resources) and he assumed that resources can only be reallocated and not donated (while we do not allow for reallocating resources but only for donating them).

For indivisible resources, assuming that agents have linear preferences over individual resources, Aziz et al. [3] studied the problem of adding/deleting a minimum number of resources from a given set of resources such that an envy-free allocation of the remaining resources exists. Our model differs from this in that we assume additive utility-based preferences, we modify a given initial allocation, and we aim for bounding the number of deleted resources and/or the decrease in social welfare. In a follow-up work to the work of Aziz et al., Dorn et al. [12] considered the goal of modifying a given set of resources such that there exists a proportional allocation, also in the setting with ordinal preferences. Notably, besides the case without an initial allocation, they also considered the variant where an initial allocation is given as in our model.

To settle up the existence of envy-free up to any good (EFX) allocations for indivisible resources, a series of works considered finding partial allocations (where some resources are allowed to remain unallocated; the unallocated resources can be interpreted as being donated) that satisfy EFX and have good qualitative and/or quantitative guarantees on the unallocated resources. Caragiannis et al. [9] showed that there always exists a partial EFX allocation whose Nash welfare is at least half of the maximum possible Nash welfare for the full set of resources. Chaudhury et al. [11] showed that there always exists a partial EFX allocation such that the number of unallocated resources is bounded by the number of agents minus one. Our goal is also to find fair partial allocations with good guarantees on the unallocated resources, but we assume that an initial allocation is already given and we study different fairness notions (EF and EF1).

To overcome that envy-free allocations often do not exist when dividing a set of indivisible resources and to generally improve existing allocations, many more approaches have been considered, including making a few resources divisible or sharable [6–8, 22, 24], subsidies and money transfers [1, 10, 19, 21], or reallocation of some resources [2, 16, 18].

1.2 Our Contributions

We study the complexity of EF/EF1 BY DONATING GOODS (EF-/EF1-DG), where given an initial allocation and two integers k^- and ℓ^+ the question is whether we can donate at most k^- resources such that the resulting allocation satisfies EF/EF1 and has utilitarian welfare at least ℓ^+ . Apart from this we also consider two special cases of this problem: EF-/EF1-DG (#) where we set ℓ^+ to zero and EF-/EF1-DG (€) where we set k^- to the number m of resources.

We split the paper into two parts. First, in Section 3, we start with the case of binary-encoded valuations. Second, in Section 4 we consider the computationally easier case with unary-encoded valuations (which is quite realistic to assume, as in most applications valuations should be “small” numbers). All hardness results for unary valuations directly imply hardness for binary valuations and all algorithms for binary valuations are also applicable to unary valuations. Moreover, notably, some of our algorithmic results from

Table 1: Overview of our results for agents with identical binary valuations. Note that all hardness results already hold for only two agents. Additionally, we prove in Proposition 3.6 that for non-identical binary valuations EF-DG (#) is W[1]-hard with respect to k^- and W[1]-hard with respect to k^+ even for only two agents.

	DG (#)	DG (€)	DG
EF1	$O(I + m \log m)$ (Th. 3.2)	NP-h. (Pr. 3.3)	$O((\ell^+)^6 + I)$ (Th. 3.4) $O((\ell^-)^6 + I)$ (Th. 3.4) W[1]-h. wrt. k^- (Pr. 3.5) NP-h. for $k^+ = 0$ (Pr. 3.3)
EF	NP-h. for $k^+ = 1$ (Th. 3.7) W[1]-h. wrt. k^- (Th. 3.7)	NP-h. for $\ell^+ = 1$ (Th. 3.7)	$O((\ell^-)^5 + I)$ (Th. 3.8)

Section 4 initially designed for unary valuations also work for binary valuations.

In Section 3, where we assume binary valuations, we mostly focus on the natural special case of identical valuations, as otherwise our problems remain computationally intractable even under quite severe restrictions. We conduct a complete parameterized analysis taking into account the following five parameters: (i) the number n of agents, (ii) the number k^- of donated resources, (iii) the number k^+ of remaining resources, (iv) the summed welfare ℓ^- of donated resources, and (v) the summed welfare ℓ^+ of the resulting allocation. An overview of our results from this part can be found in Table 1: Considering EF and identical binary valuations, we prove that our problems are NP-hard even for constant values of n , ℓ^+ , and k^+ . For the dual parameters k^- and ℓ^- , while EF-DG (#) is W[1]-hard with respect to k^- , the general problem EF-DG is solvable in time polynomial in ℓ^- . Switching from EF to EF1 leads to further tractability results: EF1-DG (#) becomes solvable in polynomial-time, while EF1-DG is now solvable in time polynomial in ℓ^- or ℓ^+ . Our results reveal several interesting contrasts. First, they suggest that for identical valuations computational problems for EF1 are easier to solve than for EF, which on a high-level is due to the fact that there always exists an EF1 sub-allocation where the least happy agents in the initial allocation get their full bundle (i.e., they keep their original bundle). Second, at least for EF1, DG (#) is easier to solve than DG (€). Third, the parameters ℓ^- and ℓ^+ seem to be more powerful than the related parameters k^- and k^+ .

In Section 4, where we assume that the agents have unary-encoded arbitrary (non-identical) valuations, in addition to the parameters considered in the first part, we also consider (vi) the maximum number d of resources allocated to an agent in the initial allocation, (vii) the maximum number w_a of resources an agent values as non-zero, (viii) the maximum number w_r of agents that value a resource as non-zero, and (ix) the maximum utility u^* that an agent assigns to a resource.¹ An overview of these results grouped by the involved parameters can be found in Tables 2 to 4 in Section 4. We discuss the results in detail in Section 4, but provide four highlights already here:

¹We do not consider these parameters in the first part because the parameter w_r is upper-bounded by the number of agents. Moreover, for the parameter u^* it makes no difference whether valuations are encoded in binary or unary. For the parameters d and w_a , our hardness results and some of our algorithmic result from the second part translate to the first part.

- We identify two relevant cases where one of our problems is fixed-parameter tractable in one of our parameters alone: EF-/EF1-DG (€) is fixed-parameter tractable with respect to d or w_a (Theorem 4.4). In contrast to this, EF-/EF1-DG (#) is NP-hard for constant $d + w_a + w_r$ for 0/1-valuations (Theorem 4.3). Notably, in all other cases, the complexity of EF-DG (€) and EF-DG (#) and of EF1-DG (€) and EF1-DG (#) are the same.
- Even for unary-encoded valuations, our problems are mostly computationally intractable. With the above mentioned exception, none of our parameters alone is sufficient to establish fixed-parameter tractability. Nevertheless, we still identify some tractable and realistic cases, for instance, when the number of agents and the maximum utility value is bounded (Proposition 4.6) or when the number/welfare of the deleted resources and the maximum number of resources assigned to an agent in the initial allocation is bounded (Proposition 4.9).
- In contrast to the binary case, the number k^- of resources to delete is a more powerful parameter than the welfare ℓ^- of the deleted resources, as for some parameter combinations involving k^- our general problem is fixed-parameter tractable but not for the respective combination involving ℓ^- .
- The number k^+ /welfare ℓ^+ of the remaining resources is a less powerful parameter than the number k^- /welfare ℓ^- of the deleted resources, because k^+/ℓ^+ constitutes a lower bound (on the remaining resources/welfare), while k^-/ℓ^- is an upper bound (on the deleted resources/welfare).

Several proofs (marked with [★]) are deferred to the full version [5].

2 PRELIMINARIES

Resource Allocation. We consider a set $A = \{a_1, \dots, a_n\}$ of n agents and a set $R = \{r_1, \dots, r_m\}$ of m resources. We assume that the agents have additive and cardinal preferences: For each agent $a \in A$ and resource $r \in R$, let $u_a(r) \in \mathbb{N}_0$ denote the utility agent a assigns to resource r . In this context, we also say that a values r as $u_a(r)$. We denote as u^* the maximum utility value an agent assigns to a resource, i.e., $u^* := \max_{a \in A, r \in R} u_a(r)$. Further, for a subset $R' \subseteq R$ of resources, we set $u_a(R') = \sum_{r \in R'} u_a(r)$. We say that agents have *binary/unary valuations* if for all $a \in A$ and $r \in R$ the utility values $u_a(r)$ in the input are encoded in binary/unary. Further, we say that agents have *0/1-valuations* if $u_a(r) \in \{0, 1\}$ for all $a \in A$ and $r \in R$.

In our parameterized analysis, we denote as $w_a \in [m]$ the maximum number of resources an agent values as non-zero, i.e., $w_a := \max_{a \in A} |\{r \in R \mid u_a(r) > 0\}|$ and as $w_r \in [n]$ the maximum number of agents that value a resource as non-zero, i.e., $w_r := \max_{r \in R} |\{a \in A \mid u_a(r) > 0\}|$.

An *allocation* π of resources to agents is a function $\pi: A \mapsto 2^R$ such that $\pi(a)$ and $\pi(a')$ are disjoint for $a \neq a' \in A$. For an agent $a \in A$ and an allocation π , $\pi(a) \subseteq R$ is the set of resources assigned to agent a in π . We also refer to $\pi(a)$ as a 's *bundle* in π . For readability, we always assume that initial allocations are complete, i.e., for every $r \in R$ there exists an agent $a \in A$ such that $r \in \pi(a)$. We refer to $\sum_{a \in A} u_a(\pi(a))$ as the (*utilitarian*) *welfare* of π .

For two agents $a \neq a' \in A$ and an allocation π , we say that agent a *envies* agent a' in π if a prefers a' 's bundle to its own bundle,

i.e., $u_a(\pi(a')) > u_a(\pi(a))$. Further, we say that agent a *envies agent a' up to one good* in π if, for all resources $r \in \pi(a')$, a prefers $\pi(a') \setminus \{r\}$ to $\pi(a)$, i.e., $\min_{r \in \pi(a')} u_a(\pi'(a) \setminus \{r\}) > u_a(\pi(a))$. An allocation is *envy-free* (EF) [*envy-free up to one good* (EF1)] if there is no agent that envies another agent [up to one good].

Envy-Freeness by Donating Goods. We now define our central problem. In the following, for clarity, as donating a resource corresponds to deleting the resource from the instance, we say that a resource is deleted if it is donated.

EF/[EF1] BY DONATING GOODS (EF-DG/[EF1-DG])

Input: Given a set A of agents, a set R of resources, an initial allocation π of resources to agents, and integers k^- and ℓ^+ .

Question: Is it possible to delete at most k^- resources from π such that the resulting allocation π' is envy-free [up to one good] and has utilitarian welfare at least ℓ^+ , i.e., $\pi'(a) \subseteq \pi(a)$ for all $a \in A$, $\sum_{a \in A} |\pi(a) \setminus \pi'(a)| \leq k^-$, and $\sum_{a \in A} u_a(\pi'(a)) \geq \ell^+$?

We also consider two special cases of this general problem. That is, the so-called number variant (EF-DG (#)/EF1-DG (#)) where we only have a bound on the number of deleted resources, i.e., $\ell^+ = 0$, and the so-called welfare variant (EF-DG (€)/EF1-DG (€)) where we only have a bound on the utilitarian welfare of the resulting allocation, i.e., $k^- = m$. For notional convenience, we write EF-DG (#/€) and EF1-DG (#/€) to refer to both the number and welfare variant.

Apart from k^- and ℓ^+ , we also consider the respective dual parameters. That is, the minimum number $k^+ := m - k^-$ of remaining resources and the maximum welfare $\ell^- := \sum_{a \in A} u_a(\pi(a)) - \ell^+$ of the deleted resources. In addition, we consider the maximum number $d := \max_{a \in A} |\pi(a)|$ of resources that an agent holds in the initial allocation π as a parameter. Throughout the paper, we assume that basic arithmetic operations (i.e., addition and subtraction) of natural numbers can be performed in constant time.

Auxiliary Problems. We introduce problems from which or to which we reduce. Given positive integers x_1, x_2, \dots, x_v , and S , SUBSET SUM is the problem to decide whether there is a set $I \subseteq [v]$ such that $\sum_{i \in I} x_i = S$. PARTITION is the SUBSET SUM problem with $S = \frac{1}{2} \sum_{i \in [v]} x_i$. Both SUBSET SUM and PARTITION are NP-hard assuming that numbers are encoded in binary [20]. The MINIMUM SIZE SUBSET SUM problem (resp. the MAXIMUM SIZE SUBSET SUM problem) is to find the size of the minimum (resp. maximum) subset whose sum is equal to S . SUBSET SUM and MINIMUM/MAXIMUM SUBSET SUM can be solved in $\mathcal{O}(vS)$ time by dynamic programming [4]. SUBSET SUM with the goal of finding a subset of size exactly k (i.e., $|I| = k$) is called k -SUM. It is W[1]-hard with respect to k [13].

We reduce some of our problems to:

MULTIPLE-CHOICE KNAPSACK

Input: A capacity c , an integer k , and ℓ sets S_i ($i \in [\ell]$), each resource $j \in S_i$ has a profit $p_{i,j}$ and a weight $w_{i,j}$.

Question: Is it possible to choose for each $i \in [\ell]$ one resource j_i from S_i such that $\sum_{i \in [\ell]} w_{i,j_i} \leq c$ and $\sum_{i \in [\ell]} p_{i,j_i} \geq k$?

The MULTIPLE-CHOICE KNAPSACK problem can be solved in $\mathcal{O}(c \cdot \sum_{i \in [\ell]} |S_i|)$ time by dynamic programming [14].

3 BINARY VALUATIONS

In this section, we assume that the valuations of agents are encoded in binary and mostly (except from Section 3.2) focus on situations

where agents have identical valuations (as we obtain NP-hardness even in this case). We start by considering EF1 (Sections 3.1 and 3.2) and afterwards turn to EF (Section 3.3).

For identical valuations, we assume that there are no resources that are valued as 0 by all agents, as otherwise we can remove them and update k^+ and m to get an equivalent instance. For identical valuations, we will call the utility function of all agents u .

3.1 EF1 and Identical Valuations

We now analyze EF1-DG and its two special cases EF1-DG (#/€) assuming that agents have identical binary valuations and identify several tractable cases for these problems. We start with two general results concerning EF1 allocations. First, we observe that as valuations are identical, an agent a is envied by another agent up to one good if a is envied by a least happy agent up to one good:

OBSERVATION 1. *For identical valuations, an allocation is EF1 if and only if the agents with the minimum utility in the allocation do not envy other agents up to one good.*

Moreover, for identical valuations, if we are given an EF1 allocation π' with $\pi'(a) \subseteq \pi(a)$ for each $a \in A$, then from this we can construct an EF1 allocation where all agents $a \in A$ get at least $\pi'(a)$ and all agents $a \in A$ that have the minimum utility in π get their full bundle $\pi(a)$ in π' . This can be done by successively adding for an agent $a \in A$ who has the minimum utility under π' and fulfills $\pi(a) \not\subseteq \pi'(a)$ an arbitrary resource from $\pi(a) \setminus \pi'(a)$.

LEMMA 3.1 (★). *Let A_0 be the set of agents that have the minimum utility in the initial allocation. For identical valuations, if there exists a solution for an instance EF1-DG (or EF1-DG (#) or EF1-DG (€)), then there is a solution such that all agents from A_0 get their full initial bundle and all other agents have higher utility than them.*

Using Observation 1 and Lemma 3.1, one can reduce EF1-DG (#) to finding for each agent $a \in A \setminus A_0$ a minimum size set $P_a \subseteq \pi(a)$ such that deleting it makes agents from A_0 no longer envy agent a up to one good. This problem can be solved using a simple greedy algorithm by moving the most valuable resources from $\pi(a)$ to P_a until all envy is resolved. This establishes that EF1-DG (#) is polynomial-time solvable for identical binary valuations:

THEOREM 3.2 (★). *For identical valuations, EF1-DG (#) can be solved in $O(|I| + m \log m)$ time.*

In contrast to this, for EF1-DG (€), solving said subproblem of finding for each agent who is not part of A_0 a subset of its initial bundle with minimum summed welfare such that deleting it makes agents from A_0 no longer envy the agent up to one good basically requires solving an instance of SUBSET SUM. Indeed, by reducing from PARTITION, we can show that in contrast to EF1-DG (#), EF1-DG (€) is NP-hard for two agents with identical binary valuations:

PROPOSITION 3.3 (★). *For identical binary valuations, EF1-DG (€) with $n = 2$ agents is NP-hard.*

Our hardness reduction from Proposition 3.3 does not have any implications on the parameterized complexity of EF1-DG (€) with respect to ℓ^+ respectively ℓ^- . In fact, parameterized by ℓ^+ respectively ℓ^- , EF1-DG (€) (and even the general problem EF1-DG) become tractable. This stands in contrast with the previously proven

NP-hardness for EF1-DG (€), which implies that EF1-DG is NP-hard for $k^+ = 0$. While this contrast between k^+ and ℓ^+ might look surprising at first glance, recall that ℓ^+ bounds the otherwise binary encoded welfare of the deleted resources. Thus, it is quite intuitive that ℓ^+ is more powerful than k^+ .

THEOREM 3.4 (★). *For identical valuations, EF1-DG can be solved in $O((\ell^+)^6 + |I|)$ or $O((\ell^-)^6 + |I|)$ time.*

PROOF FOR ℓ^+ . We begin with some pre-processing. Let $u_0 = \min_{a \in A} u(\pi(a))$. If $u_0 = 0$, in an EF1 allocation π' , each agent can hold at most one resource with non-zero value. Thus, in an optimal solution, each agent gets assigned its most valuable resource. In the following, we assume $u_0 \geq 1$. Using Theorem 3.2, we can get an EF1 allocation π' with the maximum number of resources left but without any guarantee on the welfare of π' . If $u_0 \geq \ell^+$ or $n \geq \ell^+$, then $\sum_{a \in A} u(\pi'(a)) \geq nu_0 \geq \ell^+$ is guaranteed by Lemma 3.1. In the following we assume $0 < u_0 < \ell^+$ and $n < \ell^+$. Let r^* be the resource with maximum value and a^* be the agent such that $r^* \in \pi(a^*)$. If $u(r^*) \geq \ell^+$, then either $r^* \in \pi'(a^*)$ and hence $\sum_{a \in A} u(\pi'(a)) \geq \ell^+$, or $r^* \notin \pi'(a^*)$, in which case we can replace the most valuable resource from $\pi'(a^*)$ with r^* . In both cases, we have constructed a solution. In the following, we assume that $u(r^*) < \ell^+$.

Now, we turn to the main part of the algorithm. Let A_0 be the set of agents who have the minimum utility in the initial allocation and $A^* \subseteq A \setminus A_0$ be the set of agents that are envied by agents in A_0 up to one good. According to Observation 1 and Lemma 3.1, for each $a_i \in A^*$, we need to find a set $R_i \subseteq \pi(a_i)$ such that by keeping all resources from R_i and deleting all resources from $\pi(a_i) \setminus R_i$, agent a_i will not be envied by agents in A_0 up to one good, and $\sum_{a_i \in A^*} |R_i| \geq \tilde{k}^+$ and $\sum_{a_i \in A^*} u(R_i) \geq \tilde{\ell}^+$, where $\tilde{k}^+ := k^+ - \sum_{a_i \in A \setminus A^*} |\pi(a_i)|$ and $\tilde{\ell}^+ := \ell^+ - \sum_{a_i \in A \setminus A^*} u(\pi(a_i))$. We will solve this problem in two steps. In Step 1, for each agent $a_i \in A^*$ we compute the set of all possible R_i such that after deleting $\pi(a_i) \setminus R_i$ no agent envies a_i up to one good. Then, in Step 2 we check whether it is possible to select one candidate R_i for each $a_i \in A^*$ such that $\sum_{a_i \in A^*} |R_i| \geq \tilde{k}^+$ and $\sum_{a_i \in A^*} u(R_i) \geq \tilde{\ell}^+$.

Step 1: Fix some agent $a_i \in A^*$. We want to guess the utility t of $u(R_i)$ and the utility t_1 of the most valuable resource in R_i . Let $t_2 := t - t_1$ be the utility of the remaining resources in R_i . By EF1, we have $t_2 \leq u_0 < \ell^+$. Since $t_1 \leq u(r^*) < \ell^+$, we have $t < 2\ell^+$. Thus, we can iterate over at most $2(\ell^+)^2$ different pairs (t, t_1) . For each (t, t_1) such that $t_2 = t - t_1 \leq u_0 < \ell^+$, we find an arbitrary resource $r_0 \in \pi(a_i)$ with $u(r_0) = t_1$ (if there is no such resource, then we can skip this pair), and then, we want to compute the maximum size of a subset $R_i = R'_i \cup \{r_0\}$ such that $R'_i \subseteq \pi(a_i) \setminus \{r_0\}$ and $u(R'_i) = t_2$. This is an instance of the MAXIMUM SIZE SUBSET SUM problem (see Section 2). Since $u(R'_i) = t_2 < \ell^+$, for each value $v \in [1, \ell^+]$, set R'_i can contain at most ℓ^+ resources with value v (recall that we do not have resources valued as 0). Thus, we can pick a subset $S_i \subseteq \pi(a_i) \setminus \{r_0\}$ of resources with $|S_i| \leq (\ell^+)^2$ such that we only need to include resources from S_i in R'_i . We then construct an instance of MAXIMUM SIZE SUBSET SUM with target value $t_2 < \ell^+$ and set $\{u(r) \mid r \in S_i\}$ with $|S_i| \leq (\ell^+)^2$. Using dynamic programming, the instance can be solved in $O((\ell^+)^3)$ time. We need to do this for each agent $a_i \in A^*$ and each pair (t, t_1) , separately. As we have $n < \ell^+$ agents and, as observed above,

$2(\ell^+)^2$ valid pairs, this can be done in $O((\ell^+)^6)$ time. For each agent $a_i \in A^*$, we get a family L_i of candidate sets (each such set consists of the resources selected by the dynamic program and r_0). For all candidate sets in L_i with the same utility, we just keep one candidate set of maximum cardinality. Thus, for each $t \in [1, 2\ell^+]$, there is at most one candidate in L_i and hence $|L_i| \leq 2\ell^+$.

Step 2: For each agent $a_i \in A^*$ and each $R_i^j \in L_i$, we compute a pair (s_i^j, t_i^j) , where $s_i^j = |R_i^j| \leq 2\ell^+$ and $t_i^j = u(R_i^j) \leq 2\ell^+$ resulting in a set $Q_i = \{(s_i^j, t_i^j) \mid j \in \{1, \dots, |L_i|\}\}$ of such pairs. Without loss of generality, assume $t_i^1 \geq t_i^j$ for $j \in \{1, \dots, |L_i|\}$. Now the problem is to find a pair $(s_i^{j_i}, t_i^{j_i})$ from each Q_i such that $\sum_{a_i \in A^*} s_i^{j_i} \geq \tilde{k}^+$ and $\sum_{a_i \in A^*} t_i^{j_i} \geq \tilde{\ell}^+$. This can be reduced to the MULTIPLE-CHOICE KNAPSACK PROBLEM (MCKP) (see Section 2) as follows. Let $S^* = \sum_{a_i \in A^*} s_i^1$ and $T^* = \sum_{a_i \in A^*} t_i^1$. If $T^* < \tilde{\ell}^+$, then our instance is clearly a No-instance. If $T^* \geq \tilde{\ell}^+$ and $S^* \geq \tilde{k}^+$, then we get a solution by selecting the corresponding set R_i^1 for each agent $a_i \in A^*$. Finally, if $T^* \geq \tilde{\ell}^+$ but $S^* < \tilde{k}^+$, then we need to select a pair $(s_i^{j_i}, t_i^{j_i}) \in Q_i$ for each agent $a_i \in A^*$ such that $\sum_{a_i \in A^*} (t_i^{j_i} - t_i^1) \leq T^* - \tilde{\ell}^+$ and $\sum_{a_i \in A^*} (s_i^{j_i} - s_i^1) \geq \tilde{k}^+ - S^*$. This is an instance of MCKP with sets L_i ($\forall a_i \in A^*$), capacity $T^* - \tilde{\ell}^+$, and lower bound of the target value $\tilde{k}^+ - S^*$. For each $a_i \in A^*$, each set $R_i^j \in L_i$ has weight $t_i^1 - t_i^j$ and value $s_i^j - s_i^1$. Since $\sum_{a_i \in A^*} |L_i| \leq 2(\ell^+)^2$ and $T^* - \tilde{\ell}^+ \leq T^* \leq |A^*| \cdot \max_{a_i \in A^*} t_i^1 \leq 2(\ell^+)^2$ the instance can be solved in $O((\ell^+)^4)$ time [14]. Summing up, our problem can be solved in $O((\ell^+)^6 + |I|)$ time. \square

While we have seen in Theorem 3.2 that EF1-DG (#) is polynomial-time solvable, we now show that EF1-DG parameterized by k^- is W[1]-hard even for only two agents, which stands in contrast to the preceding tractability results for ℓ^- :

PROPOSITION 3.5 (★). *For identical binary valuations, EF1-DG with $n = 2$ agents is W[1]-hard parameterized by k^- .*

3.2 EF1 and Non-Identical Preferences

Our positive result for EF1-DG (#) from Theorem 3.2 for identical binary valuations raises the question whether there is hope for tractability results for EF1-DG (#) parameterized by n , k^- , or k^+ with non-identical binary valuations. We answer this question negatively with a strong (parameterized) hardness result in the following proposition:

PROPOSITION 3.6 (★). *For binary valuations, EF1-DG (#) with $n = 2$ agents is W[1]-hard parameterized by k^+ or k^- , even if the two agents only disagree on the valuation of two resources.*

3.3 EF and Identical Valuations

We now turn to EF and start by proving a strong hardness result for EF-DG (#/€): Reducing from SUBSET SUM, we show that even if we only have two agents with identical valuations, EF-DG (#/€) is NP-hard. Even stronger, EF-DG (#/€) is NP-hard even if we are allowed to delete all but one resource.

THEOREM 3.7 (★). *For identical binary valuations, EF-DG (#) with $k^+ = 1$ and EF-DG (€) with $\ell^+ = 1$ are NP-hard for $n = 2$ agents, and EF-DG (#) with $n = 2$ agents is W[1]-hard parameterized by k^- .*

This strong hardness result puts EF1 and EF in a sharp contrast, as EF1-DG (#) is polynomial-time solvable and EF1-DG is solvable in time polynomial in ℓ^+ . Concerning the later contrast, on a high level, the reason why ℓ^+ is a more powerful parameter for EF1-DG than for EF-DG is that we know by Lemma 3.1 that there always is an EF1 allocation where all agents hold a bundle of utility at least $\min_{a \in A} u(\pi(a))$. Thus, for “small” values of ℓ^+ we can return yes, while “larger” values of ℓ^+ allow for a tractable algorithm in ℓ^+ .

However again similar to the EF1 case, using a simplified version of Step 1 from Theorem 3.4, in the following theorem, we show that in contrast to the W[1]-hardness with respect to k^- and NP-hardness for $\ell^+ = 1$ from the previous theorem, EF-DG (€) (and, in fact, even EF-DG) is solvable in time polynomial in ℓ^- .

THEOREM 3.8 (★). *For identical valuations, EF-DG can be solved in $O((\ell^-)^5 + |I|)$ time.*

4 UNARY VALUATIONS

In this section, we consider unary-encoded valuations and conduct a thorough parameterized complexity analysis of our problems with respect to various natural and problem-specific parameters. We start by proving that all our problems are NP-hard even for 0/1-valuations by establishing a simple reduction from SET COVER to EF-DG (#/€)/EF1-DG (#/€):

THEOREM 4.1 (★). *EF-DG (#/€) and EF1-DG (#/€) are NP-hard for 0/1-valuations.*

This hardness result motivates us to explore the parameterized complexity of our problems.² We start by considering in Section 4.1 our problems in case that valuations and the initial allocation are “sparse”. Subsequently, in Section 4.2, we consider the influence of the number of agents on the problem. Afterwards, in Section 4.3, we turn to the problem specific parameter k^- and ℓ^- , i.e., the maximum number/welfare of the deleted resources. Lastly, in Section 4.4, we consider the dual parameters k^+ and ℓ^+ that quantify the minimum number/welfare of remaining resources. We also consider combined parameters, where we mostly include results for a combined parameter in the latest section regarding one of the parameters from the combination. We refer to Table 2 for an overview of results from Sections 4.1 and 4.2, to Table 3 for an overview of results from Section 4.3, and to Table 4 for an overview of results from Section 4.4.

4.1 Sparse Valuations and Allocations

In this section, we consider instances where the initial allocation is sparse, i.e., the maximum number d of resources assigned to an agent in the initial allocation is bounded, and valuations are sparse, i.e., the maximum number w_a of resources that an agent values as non-zero and the maximum number w_r of agents that value a specific resource as non-zero are bounded (see Table 2 for an overview of our results from this and the next section). We prove that EF-DG (#) and EF1-DG (#) are NP-hard even for constant value of $d + w_a + w_r$ and 0/1-valuations. In a sharp contrast to this, we

²Note that in the reduction from Theorem 4.1, we set the number/welfare of the deleted resources k^-/ℓ^- to be the requested size z of the set cover. As SET COVER is W[2]-hard parameterized by z , the reduction already establishes that EF-DG (#/€) and EF1-DG (#/€) are W[2]-hard for k^-/ℓ^- .

Table 2: Overview of our results for parameters n , w_a , w_r , and d and their combinations.

EF-DG (#) EF1-DG (#)	EF-DG (€) EF1-DG (€)	EF-DG EF1-DG
NP-h. for $d = 3/d = 7$, $w_a = 4/w_a = 7$, $w_r = 3$, and 0/1-val. (Th. 4.3)	NP-h. for $w_r = 4$ and 0/1 val. (Co. 4.2) FPT wrt. d , FPT wrt. w_a (Th. 4.4)	
W[1]-h. wrt. $n + k^-$ (Th. 4.5)	W[1]-h. wrt. $n + \ell^-$ (Th. 4.5)	FPT wrt. $u^* + n$ (Pr. 4.6) FPT wrt. $d + n$ (Ob. 2) FPT wrt. $w_a + n$ (Ob. 2)

show that EF-DG (€) and EF1-DG (€) are fixed-parameter tractable with respect to w_a or d for arbitrary valuations (and NP-hard for a constant value of w_r and 0/1-valuations).

First of all note that one can reduce RESTRICTED EXACT COVER BY 3-SETS, which is an NP-hard special case of SET COVER where each set consists of exactly three elements and each element appears in exactly three sets [17, Problem SP2] to our problems (by applying the reduction from Theorem 4.1 but reducing from RESTRICTED EXACT COVER BY 3-SETS instead of SET COVER). This allows us to establish the following:

COROLLARY 4.2. *EF-DG (#)/EF1-DG (#) is NP-hard for 0/1-valuations even if $w_a = 5/w_a = 4$ and $w_r = 3$. EF-DG (€)/EF1-DG (€) is NP-hard for 0/1-valuations even if $w_r = 4$.*

Indeed, using a more involved reduction, one can show that for EF-DG (#)/EF1-DG (#) this para-NP hardness result can be extended to also include the maximum number d of resources an agent holds in the initial allocation:

THEOREM 4.3 (★). *EF-DG (#) is NP-hard for 0/1-valuations, even if $d = 3$, $w_a = 4$, and $w_r = 3$. EF1-DG (#) is NP-hard for 0/1-valuations, even if $d = 7$, $w_a = 7$, and $w_r = 3$.*

In contrast to this, we now show that EF-DG (€) and EF1-DG (€) are fixed-parameter tractable with respect to d or w_a for arbitrary (even binary) valuations. This result is surprising in two ways. First, we have proven in the previous theorem that EF-DG (#) and EF1-DG (#) are NP-hard even for a constant value of $d + w_a + w_r$ and 0/1-valuations. Notably, this is our first and only case for unary valuations where one of EF-DG (#) and EF-DG (€) (or EF1-DG (#) and EF1-DG (€)) is harder than the other. Second, we have already seen that EF-DG (€) and EF1-DG (€) are NP-hard for a constant value of the closely related parameter w_r for 0/1-valuations.

THEOREM 4.4 (★). *EF-DG (€)/EF1-DG (€) are solvable in $O(2^{2d} \cdot d \cdot n^2 + |I|)$ time and solvable in $O(2^{2w_a} \cdot w_a \cdot n^2 + |I|)$ time.*

BEGINNING OF PROOF. We only describe our approach for EF-DG (€). First, we delete all resources that are valued as zero by the agent holding them in the initial allocation and denote the resulting allocation by π_0 . Next, as long as there is an agent a that is envied by some agent, we compute a subset $S_a \subseteq \pi_0(a)$ of resources with the highest utility for a such that a will not be envied after we delete all resources in $\pi_0(a) \setminus S_a$. Subsequently, we set a 's bundle to be S_a (but do not update $\pi_0(a)$). Some agent may become envied by other agents multiple times over the execution of the algorithm. Each time, we recompute the optimal subset S_a from $\pi_0(a)$. This implies, for instance, that an agent might get back once deleted resources.

Let π^* be the resulting allocation. It is clear that π^* is envy-free. To show that π^* has the highest utilitarian welfare among all envy-free allocations, we show an even stronger result. We say an allocation (which is not necessarily envy-free) is *maximal* if every agent has higher or equal utility in this allocation than in every envy-free allocation that can result from deleting resources from the initial allocation π . We show that π^* is maximal by induction. First of all, π_0 is maximal, as every agent has the same utility as in the initial allocation. Next, we show that if in some step the current allocation π_1 is maximal, then after this step the resulting allocation π_2 is also maximal. Let a be the agent whose bundle has been changed in this step. Since π_1 is maximal and a is the only agent with $\pi_1(a) \neq \pi_2(a)$, it suffices to show that for each envy-free allocation π' , we have $u_a(\pi_2(a)) \geq u_a(\pi'(a))$. Suppose for contradiction that $u_a(\pi_2(a)) < u_a(\pi'(a))$. Then, we modify π_2 by giving agent a the bundle $S'_a := \pi'(a) \cap \pi_0(a)$ and get a new allocation π'_2 (where by our above assumption agent a has a higher utility in π'_2 than in π_2). We claim that in π'_2 agent a is not envied by other agents. Indeed, since π_1 is maximal and every agent $a' \in A \setminus \{a\}$ has the same set of resources in π'_2 , π_2 , and π_1 , we have that agent a' has higher or equal utility in π'_2 as in π' , i.e., $u_{a'}(\pi'_2(a')) = u_{a'}(\pi_2(a')) = u_{a'}(\pi_1(a')) \geq u_{a'}(\pi'(a))$. Moreover, since π' is envy-free, which means that a is not envied by other agents in π' , and $\pi'_2(a) = S'_a \subseteq \pi'(a)$, we get that a is not envied by other agents in π'_2 . However, this contradicts the choice of $\pi_2(a)$ because S'_a is a set of higher utility than $\pi_2(a)$ such that no agent envies a . Thus, π^* is maximal. \square

4.2 Number of Agents

Having seen that at least for EF-DG (#) and EF1-DG (#) sparse valuations and allocations do not help in our search for tractable cases, we now turn to the number n of agents. Before we identify two tractable cases, we start by showing that our problems are W[1]-hard with respect to the number of agents even considered in combination with the number (welfare) of deleted resources (we will examine the parameters number and welfare of deleted resources in more detail in the next subsection).

THEOREM 4.5 (★). *For unary valuations, EF-DG (#/€) and EF1-DG (#/€) are W[1]-hard parameterized by $n + k^-/n + \ell^-$.*

On the positive side, combining the number of agents with the maximum utility value an agent assigns to a resource, our general problems can be encoded in an ILP where the number of constraints is quadratic in n . Subsequently, we can employ the algorithm by Eisenbrand and Weismantel [15]:

PROPOSITION 4.6 (★). *EF-DG and EF1-DG are solvable in $(n^2 u^*)^{O(n^2)} \cdot m^2$ time.*

The previous result implies that our general problem for both EF and EF1 is in FPT with respect to $u^* + n$, in XP with respect to n , and FPT with respect to n for 0/1-valuations.

Examining now the combination of the number n of agents with the parameters introduced in Section 4.1, first note that n upper-bounds the number w_r of agents that value a resource as non-zero. However, if we combine the number n of agents with the maximum number w_a of resources that an agent values as non-zero or the maximum number d of resources an agent holds in

Table 3: Overview of our results for parameters k^- and ℓ^- and all parameter combinations they are involved in.

EF-DG (#) EF1-DG (#)	EF-DG (€) EF1-DG (€)	EF-DG EF1-DG
W[2]-h. wrt. k^- for 0/1-val. (Co. 4.7)	XP wrt. ℓ^- W[2]-h. wrt. ℓ^- for 0/1-val. (Co. 4.7)	XP wrt. k^- NP-h. for $w_a = 5/w_a = 4$, $w_r = 3$, $\ell^- = 0$, and 0/1-val. (Co. 4.8)
W[1]-h. wrt. $n + k^-$ (Th. 4.5)	W[1]-h. wrt. $n + \ell^-$ (Th. 4.5)	
	FPT wrt. d (Th. 4.4) FPT wrt. w_a (Th. 4.4) FPT wrt. $u^* + w_r + \ell^-$ (Pr. 4.11)	FPT wrt. $d + k^-$ (Pr. 4.9) XP wrt. $d + \ell^-$ (Pr. 4.10) W[1]-h. wrt. ℓ^- for $d = 4/d = 5$ (Pr. 4.10) FPT wrt. $w_a + k^-$ (Pr. 4.9) FPT wrt. $u^* + w_r + k^-$ (Pr. 4.11)

the initial allocation, then we can bound the number of “relevant” resources and thereby the size of the whole instance in a function of the combined parameter $n + w_a$ or $n + d$:

OBSERVATION 2 (★). *EF-DG and EF1-DG are solvable in $O(|I| + n^2 \cdot m \cdot 2^{n \cdot w_a})$ time and solvable in $O(|I| + n^2 \cdot m \cdot 2^{n \cdot d})$ time.*

4.3 Number/Welfare of Deleted Resources

In this section, we examine the influence of the number/welfare k^-/ℓ^- of resources to be deleted and identify some tractable cases (see Table 3 for an overview of our results). On the hardness side, in Theorem 4.5, we have already shown that this parameter (even in combination with the number of agents) is not sufficient to lead fixed-parameter tractability (unless $\text{FPT}=\text{W}[1]$). Moreover, in Theorem 4.1 we have constructed a (parameterized) reduction from the W[2]-hard SET COVER problem to our problems which also establishes the following:

COROLLARY 4.7. *Parameterized by k^-/ℓ^- , EF-DG (#/€) and EF1-DG (#/€) are W[2]-hard for 0/1-valuations.*

On the algorithmic side, there is an XP algorithm with respect to parameter k^- for EF-DG and EF1-DG running in time $O(|I| + m^{k^-} \cdot k^- \cdot n^2)$ by simply iterating over all size- k^- subsets of resources and checking whether the allocation that results from deleting these resources is envy-free (up to one good). For the parameter ℓ^- , there also is an XP algorithm running in $O(|I| + m^{\ell^-} \cdot \ell^- \cdot n^2)$ for EF-DG (€) and EF1-DG (€) by first deleting all resources that are valued as zero by the agent holding it (which never creates any envy) and subsequently iterating over all size- ℓ^- subsets of resources and checking whether additionally deleting these resources makes the allocation envy-free (up to one good). However, in the first step of the XP algorithm for ℓ^- , an arbitrary number of resources might get deleted which raises the question whether there is an XP algorithm for the parameter ℓ^- for the general problems. In fact, using the construction from Theorem 4.1 for EF-DG (#)/EF1-DG (#) and reducing from RESTRICTED EXACT COVER BY 3-SETS while setting $k^- = z$ and $\ell^- = 0$ rules out this possibility:

COROLLARY 4.8. *EF-DG/EF1-DG is NP-hard for 0/1-valuations even if $\ell^- = 0$, $w_a = 5/w_a = 4$, and $w_r = 3$.*

We now consider k^-/ℓ^- in combination with sparse valuations or a sparse initial allocation and design several fixed-parameter

tractable algorithms. Using a search-tree approach, we show fixed-parameter tractability of the parameter combination $d + k^-/d + \ell^-$ and $w_a + k^-/w_a + \ell^-$ for arbitrary (even binary) valuations (notably, we have already proven in Theorem 4.4 that EF-DG (€)/EF1-DG (€) is fixed parameter tractable in d or w_a):

PROPOSITION 4.9 (★). *EF-DG/EF1-DG is solvable in $O(|I| + n \cdot d^{k^-})$ time and solvable in $O(|I| + n \cdot w_a^{k^-})$ time. EF-DG (€)/EF1-DG (€) is solvable in $O(|I| + n \cdot d^{\ell^-})$ time and solvable in $O(|I| + n \cdot w_a^{\ell^-})$ time.*

As already observed above for the XP algorithm, here again ℓ^- in contrast to k^- is not enough to establish an algorithmic result for the general problem. Observe that we have already seen in Corollary 4.8 that both general problems are NP-hard for constant w_a and constant ℓ^- . Moreover, we show in the following proposition using a slightly involved reduction from CLIQUE that the FPT algorithm for $d + \ell^-$ cannot be extended to the general problem (like for $d + k^-$). However, EF-DG/EF1-DG parameterized by $d + \ell^-$ is in XP for arbitrary (binary or unary) valuations.

PROPOSITION 4.10 (★). *EF-DG/EF1-DG is solvable in $O(|I| + m^{\ell^-} \cdot 2^d \cdot m \cdot n)$ time. For unary valuations, parameterized by ℓ^- , EF-DG/EF1-DG is W[1]-hard even if $d = 4/d = 5$.*

Lastly, we turn to the combined parameter k^- plus the maximum number of agents w_r that value a resource as non-zero. As the number n of agents upper-bounds w_r , as proven in Theorem 4.5, this parameter combination is not enough to achieve fixed-parameter tractability. However, by adding the maximum utility value, tractability can be regained:

PROPOSITION 4.11 (★). *EF-DG and EF1-DG are solvable in $O((u^* + 1)^{k^- \cdot (w_r + 1)} \cdot k^- \cdot |I|)$ time. EF-DG (€) and EF1-DG (€) are solvable in $O((u^* + 1)^{\ell^- \cdot (w_r + 1)} \cdot \ell^- \cdot |I|)$ time.*

PROOF SKETCH FOR EF AND PARAMETER k^- . Our algorithm crucially relies on the simple observation that by deleting k^- resources the envy of at most $k^- \cdot w_r$ agents can be resolved. Assuming that we are given a valid solution where we delete at most k^- resources one after each other, let $A' \subseteq A$ be the set of agents from whose bundle a resource got deleted or that envy another agent at some point. From our above observation it follows that $|A'| \leq k^- \cdot (w_r + 1)$. Assuming that we would know A' we could as long as we still have budget left and there is an agent a that envies another agent a' guess the utility profile $(u_b(r))_{b \in A'}$ of the resource $r \in \pi(a')$ to be deleted from a' 's bundle restricted to A' (there exist $u^{*k^- \cdot (w_r + 1)}$ such guesses) and subsequently delete this resource. However, unfortunately, we do not know the set A' of agents upfront which makes it necessary to update this set of agents on the fly and complete the previous guesses concerning the utility profiles of deleted resources. We do so by initializing A' with the set of agents that initially envy another agent. Then, as long as there is an agent $a \in A'$ envying another agent a' , which implies that a resource from a' 's bundle needs to be deleted, we guess the utility which agents from A' have for the resource to be deleted from a' 's bundle and adjust the utility that these agents have for a' 's bundle. Moreover, if a' is not already part of A' , then we add a' to A' and guess the value that a' has for all already “deleted” resources and adjust a' 's valuations

Table 4: Overview of our results for parameters k^+ and ℓ^+ and parameter combinations involving them.

EF-DG (#/€)	EF1-DG (#)	EF1-DG (€)
NP-h. for $k^+ = 1/\ell^+ = 1$ and $w_r = 4$ (Th. 4.12)	W[1]-h. wrt. k^+ for $w_a = 2$ and 0/1 val. (Pr. 4.14)	W[1]-h. wrt ℓ^+ for 0/1 val. (Pr. 4.14)
FPT wrt. k^+/ℓ^+ for 0/1-val. (Pr. 4.13)	FPT wrt. $k^+ + d$ (Pr. 4.15)	FPT wrt. $\ell^+ + d$ (Pr. 4.15)

accordingly. Finally, we check whether there in fact exists a set of at most k^- resources satisfying our guesses. \square

Notably, Corollary 4.8 rules out the possibility that the FPT-algorithm from above for EF-DG (€) and EF1-DG (€) for parameters $\ell^- + u^* + w_r$ can be extended to the respective general problem.

4.4 Number/Welfare of Remaining Resources

In this section, we examine the influence of the number/welfare k^+/ℓ^+ of resources that remain after the deletion of resources (see Table 4 for an overview of our results). We show that, generally speaking, these parameters are less powerful than the respective dual parameters but we nevertheless identify two tractable cases. We also show an interesting contrast between EF and EF1.

We start by proving a very strong intractability result for EF, namely, that deciding whether there is a solution that does not delete all resources is already NP-hard:

THEOREM 4.12 (★). *For unary valuations, EF-DG (#/€) is NP-hard even if $k^+ = 1/\ell^+ = 1$ and $w_r = 4$.*

PROOF SKETCH. We reduce from INDEPENDENT SET on cubic graphs, i.e., graphs where each vertex has exactly three neighbors. Given a cubic graph $G = (V, E = \{e_1, \dots, e_q\})$ and an integer t , the task is to decide whether G contains an independent set of size t . We first add a *special agent* a^* and for each $i \in [q]$ an *edge agent* a_{e_i} . Moreover, for each $v \in V$, we add a *vertex resource* r_v , which is valued as one by the special agent and by all edge agents corresponding to edges incident to v . We allocate all vertex resources to the special agent. Furthermore, for $i \in [q]$, we add a resource r_{e_i} which we allocate to a_{e_i} and which is valued as one by a_{e_i} and by $a_{e_{i-1 \bmod q}}$ and as t by the special agent. Lastly, we set $k^+ = 1/\ell^+ = 1$.

One can show that after the deletion of resources, each edge agent needs to hold its resource and the special agent needs to hold at least t resources. Then, the vertices $V' \subseteq V$ corresponding to the vertex resources the special agent holds form an independent set, as the existence of an edge $e_i \in E$ with both endpoints in V' would imply that a_{e_i} envies the special agent. \square

Note that the above reduction crucially relies on the fact that the special agent values some resources as t . In fact, if we have 0/1-valuations, we can establish fixed-parameter tractability for the number and welfare (but not the general) problem by first deleting all agents with zero welfare in the initial allocation and all resources they value as non-zero and then making a case distinction based on whether k^+/ℓ^+ is larger than the remaining number of agents:

PROPOSITION 4.13 (★). *EF-DG (#/€) is solvable in $|I| + (k^+)^2 \cdot m^2 \cdot |I| + (\ell^+)^2 \cdot m^2$ time for 0/1-valuations.*

In contrast to this result for EF, EF1-DG (#/€) parameterized by k^+/ℓ^+ is W[1]-hard for 0/1-valuations. We use a reduction similar to the one used in Theorem 4.12 but reduce from INDEPENDENT SET and delete all resources held by edge agents and set $k^+ = t/\ell^+ = t$.

PROPOSITION 4.14 (★). *Parameterized by k^+ , EF1-DG (#) is W[1]-hard for 0/1-valuations, even if $w_a = 2$ and the initial allocation has zero utilitarian welfare. Parameterized by ℓ^+ , EF1-DG (€) is W[1]-hard for 0/1-valuations.*

It is an intriguing open question whether EF1-DG (#/€) is in XP with respect to k^+/ℓ^+ or para-NP-hard. Note that an XP algorithm for unary valuations would be quite surprising, as this would imply that for the parameter k^+/ℓ^+ , for unary valuations, EF1 is easier than EF, while, for 0/1-valuations, EF is easier than EF1.

Lastly, by making a case distinction based on whether more or less than k^+/ℓ^+ agents hold a resource in the initial allocation, we show that EF1-DG (#/€) is fixed-parameter tractable with respect to $k^+ + d/\ell^+ + d$ for arbitrary (unary or binary) valuations:

PROPOSITION 4.15 (★). *EF1-DG (#/€) is solvable in $O(|I| + 2^{k^+ \cdot d} \cdot m \cdot n^2)/O(|I| + 2^{\ell^+ \cdot d} \cdot m \cdot n^2)$ time.*

While our picture for the other parameters is nearly complete, there exist various open questions for parameters k^+ and ℓ^+ . For instance, the complexity of our problems parameterized by $n + k^+$, $n + \ell^+$, $d + k^+$, $d + \ell^+$, $w_a + k^+$, and $w_a + \ell^+$ is open.

5 CONCLUSION

We studied the complexity of making an initial allocation envy-free by donating a subset of resources satisfying certain constraints. While we have shown that this problem is NP-hard even under quite severe restrictions on the input, resorting to parameterized complexity theory, we identified numerous tractable cases. Moreover, we discovered several interesting contrasts between seemingly closely related parameters and problem variants. For future work, it would be possible to change the considered fairness criterion and to consider, for example, proportionality or maximin share. Furthermore, instead of or in addition to allowing that some resources are donated, one could also analyze reallocating some resources, which seems to be a harder setting as many of our intractability results (such as Theorem 4.1) can be converted to this setting by adding some dummy agents to “receive” donated resources. Lastly, from a more practical point of view, it would be interesting to experimentally measure how many resources would need to be donated to make allocations envy-free that are computed using an algorithm or heuristic which tries to compute an allocation with few envy.

ACKNOWLEDGMENTS

NB was supported by the DFG project MaMu (NI 369/19) and by the DFG project ComSoc-MPMS (NI 369/22). KH was supported by the DFG Research Training Group 2434 “Facets of Complexity” and by the DFG project FPTinP (NI 369/16). DK acknowledges the support of the OP VVV MEYS funded project CZ.02.1.01/0.0/0.0/16_019/0000765 “Research Center for Informatics”. JL was supported by the DFG project “AFFA” (BR 5207/1; NI 369/15) and the Ministry of Education, Singapore, under its Academic Research Fund Tier 2 (MOE2019-T2-1-045).

REFERENCES

- [1] Haris Aziz. 2021. Achieving Envy-freeness and Equitability with Monetary Transfers. In *Proceedings of the Thirty-Fifth AAAI Conference on Artificial Intelligence (AAAI '21)*. AAAI Press, 5102–5109.
- [2] Haris Aziz, Péter Biró, Jérôme Lang, Julien Lesca, and Jérôme Monnot. 2016. Optimal Reallocation under Additive and Ordinal Preferences. In *Proceedings of the 2016 International Conference on Autonomous Agents and Multiagent Systems (AAMAS '16)*. ACM, 402–410.
- [3] Haris Aziz, Ildikó Schlotter, and Toby Walsh. 2016. Control of Fair Division. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence (IJCAI '16)*. IJCAI/AAAI Press, 67–73.
- [4] Richard Bellman. 1956. Notes on the theory of dynamic programming IV - Maximization over discrete sets. *Naval Research Logistics Quarterly* 3, 1-2 (1956), 67–70.
- [5] Niclas Boehmer, Robert Bredereck, Klaus Heeger, Dušan Knop, and Junjie Luo. 2022. Multivariate Algorithmics for Eliminating Envy by Donating Goods. *CoRR* abs/2202.01716 (2022). arXiv:2202.01716 <https://arxiv.org/abs/2202.01716>
- [6] Steven J. Brams, D. Marc Kilgour, and Christian Klamler. 2014. Two-person fair division of indivisible items: An efficient, envy-free algorithm. *Notices of the American Mathematical Society* 61, 2 (2014), 130–141.
- [7] Steven J Brams and Alan D Taylor. 1996. A procedure for divorce settlements. *Mediation Quarterly* 13, 3 (1996), 191–205.
- [8] Robert Bredereck, Andrzej Kaczmarczyk, Junjie Luo, Rolf Niedermeier, and Florian Sachse. 2022. On Improving Resource Allocations by Sharing. In *Proceedings of the 36th AAAI Conference on Artificial Intelligence (AAAI '22)*. <https://arxiv.org/abs/2112.07525> Accepted for publication.
- [9] Ioannis Caragiannis, Nick Gravin, and Xin Huang. 2019. Envy-Freeness Up to Any Item with High Nash Welfare: The Virtue of Donating Items. In *Proceedings of the 2019 ACM Conference on Economics and Computation (EC '19)*. ACM, 527–545.
- [10] Ioannis Caragiannis and Stavros Ioannidis. 2021. Computing Envy-Freeable Allocations with Limited Subsidies. In *Proceedings of 17th International Conference on Web and Internet Economics (WINE '21)*. Springer, 522–539.
- [11] Bhaskar Ray Chaudhury, Telikepalli Kavitha, Kurt Mehlhorn, and Alkmini Sgouritsa. 2020. A Little Charity Guarantees Almost Envy-Freeness. In *Proceedings of the 2020 ACM-SIAM Symposium on Discrete Algorithms, (SODA '20)*. SIAM, 2658–2672.
- [12] Britta Dorn, Ronald de Haan, and Ildikó Schlotter. 2021. Obtaining a Proportional Allocation by Deleting Items. *Algorithmica* 83, 5 (2021), 1559–1603.
- [13] Rodney G. Downey and Michael R. Fellows. 1995. Fixed-Parameter Tractability and Completeness II: On Completeness for W[1]. *Theoretical Computer Science* 141, 1&2 (1995), 109–131.
- [14] Krzysztof Dudziński and Stanisław Walukiewicz. 1987. Exact methods for the knapsack problem and its generalizations. *European Journal of Operational Research* 28, 1 (1987), 3–21.
- [15] Friedrich Eisenbrand and Robert Weismantel. 2020. Proximity Results and Faster Algorithms for Integer Programming Using the Steinitz Lemma. *ACM Transactions on Algorithms* 16, 1 (2020), 5:1–5:14.
- [16] Ulle Endriss. 2013. Reduction of economic inequality in combinatorial domains. In *Proceedings of the 12th International Conference on Autonomous Agents and Multiagent Systems (AAMAS '13)*. IFAAMAS, 175–182.
- [17] M. R. Garey and David S. Johnson. 1979. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman.
- [18] Laurent Gourvès, Julien Lesca, and Anaëlle Wilczynski. 2017. Object Allocation via Swaps along a Social Network. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence (IJCAI '17)*. ijcai.org, 213–219.
- [19] Daniel Halpern and Nisarg Shah. 2019. Fair Division with Subsidy. In *Proceedings of the 12th International Symposium on Algorithmic Game Theory (SAGT '19)*. Springer, 374–389.
- [20] Richard M. Karp. 1972. Reducibility Among Combinatorial Problems. In *Proceedings of a symposium on the Complexity of Computer Computations, held March 20-22, 1972, at the IBM Thomas J. Watson Research Center (The IBM Research Symposia Series)*. Plenum Press, 85–103.
- [21] Vishnu V. Narayan, Mashbat Suzuki, and Adrian Vetta. 2021. Two Birds With One Stone: Fairness and Welfare via Transfers. *CoRR* abs/2106.00841 (2021). arXiv:2106.00841 <https://arxiv.org/abs/2106.00841> Accepted at SAGT '21.
- [22] Fedor Sandomirskiy and Erel Segal-Halevi. 2019. Fair Division with Minimal Sharing. *CoRR* abs/1908.01669 (2019). arXiv:1908.01669 <http://arxiv.org/abs/1908.01669>
- [23] Erel Segal-Halevi. 2018. Redividing the Cake. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence (IJCAI '18)*. ijcai.org, 498–504.
- [24] Rudolf Vetschera and D. Marc Kilgour. 2013. Strategic Behavior in Contested-Pile Methods for Fair Division of Indivisible Items. *Group Decision and Negotiation* 22, 2 (2013), 299–319.