Distribution Rules Under Dichotomous Preferences: Two Out of Three Ain't Bad

FLORIAN BRANDL, Princeton University, USA
FELIX BRANDT, Technical University of Munich, Germany
DOMINIK PETERS, Harvard University, USA
CHRISTIAN STRICKER, Technical University of Munich, Germany

We consider a setting in which agents contribute amounts of a divisible resource (such as money or time) to a common pool, which is used to finance projects of public interest. How the collected resources are to be distributed among the projects is decided by a distribution rule that takes as input a set of approved projects for each agent. An important application of this setting is donor coordination, which allows philanthropists to find an efficient and mutually agreeable distribution of their donations. We analyze various distribution rules (including the Nash product rule and the conditional utilitarian rule) in terms of classic as well as new axioms, and propose the first fair distribution rule that satisfies efficiency and monotonicity. Our main result settles a long-standing open question of Bogomolnaia, Moulin, and Stong (2005) by showing that no strategyproof and efficient rule can guarantee that at least one approved project of each agent receives a positive amount of the resource. The proof reasons about 386 preference profiles and was obtained using a computer-aided method involving SAT solvers.

1 INTRODUCTION

The question we study in this paper is how a common pool of some divisible resource ought to be distributed among a set of public projects based on the preferences of multiple agents. Following a model introduced by Bogomolnaia et al. [2005], we will study distribution rules with preferences specified as *approvals*: each agent reports an arbitrary subset of approved projects. We assume that agents prefer distributions where a greater fraction is allocated to approved projects. Approval voting is one of the most common ways of eliciting preferences in polls and online tools, due to its ease of use while still being reasonably expressive. It also naturally comes with an assumption of dichotomous preference that makes the mechanism design problem analytically tractable and thereby more theoretically productive.

Approval-based distribution rules are useful for many applications, and one can interpret the outcome distribution in at least four distinct ways.

Randomization. The obvious interpretation of probability distributions are lotteries. Rather than using a deterministic rule to pick a single proejct, one may use randomization to achieve fairness guarantees and strong participation incentives *ex ante*.

Repeated decisions. When the same decision situation occurs repeatedly, we may want to alternate between different projects to ensure fairness. For example, the organizers of a seminar may poll participants about their preferences for the day of the week when the seminar is scheduled. The best decision may be a mixture, where 10% of the seminars are held on Wednesdays, 50% on Thursdays, and 40% on Fridays.

Budget division. An organization has to decide over the use of its budget, and how to divide it among different projects. It can then let its members vote over which projects are worthy of receiving funding. The result is a distribution of the budget to projects. The budget need not be monetary, and we could divide a time budget using the same methods. For example, imagine an instructor who asks the students in the class which topics they are interested in, and who divides the class time among topics using a distribution rule.

Donor coordination. A set of donors intends to give money to charity. Each donor has a collection of favored charities that they are willing to donate to. Instead of directing their donations individually, the donors can pool their contributions and use a voting rule to divide the pool among charities. With a suitably chosen distribution rule, the result may be preferable to uncoordinated donations, in the eyes of every participant. The simplest example is if one donor favors charities a and b, while another favors b and c. An efficient voting rule will decide to direct both donors' contributions to b, which means that in the eyes of either donor, more money has been directed to a favored charity than if each had decided to split their money evenly between their own favored charities.

With these applications in mind, we will study distribution rules via an axiomatic analysis. We will look at Pareto efficiency, robustness to strategic misrepresentation, monotonicity, participation, and fairness properties. We build on previous work in this model [Aziz et al., 2019, Bogomolnaia et al., 2005, Duddy, 2015], and begin our discussion in Sections 3 and 4 by presenting the axioms and rules that have been studied in the literature. We also introduce a new rule and some additional axioms inspired by the applications mentioned above.

Compared to the usual situation in social choice, our approval-based model allows for voting rules with strong axiomatic guarantees. For example, the *utilitarian rule* (*UTIL*) which maximizes utilitarian social welfare is both efficient and strategyproof. However, it is not fair: the rule will spend the entire budget on projects that are approved by the highest number of people. There may be a large fraction of the agents who do not approve of these popular projects, and which are left unrepresented in the budget division. To fix this flaw, the *conditional utilitarian rule* (*CUT*) maximizes utilitarian welfare subject to a fairness constraint. The resulting rule is fair by definition, and it remains strategyproof. However, it fails to be efficient. A third rule, the *Nash rule* (*NASH*) which maximizes the product of agent utilities, combines fairness with efficiency, but violates strategyproofness.

Our main contribution is an impossibility theorem that answers a long-standing open question raised by Bogomolnaia et al. [2002, 2005] in the paper introducing the model of approval-based distribution rules. They ask whether the three properties in Table 1—efficiency, strategyproofness, fairness—can be satisfied together. To formalize fairness, they introduced an extremely weak property called *positive share*, which merely requires that at least one approved project of each agent receives a positive amount of the resource. Bogomolnaia et al. [2005] wrote that they "submit as a challenging conjecture the following statement: there is no strategyproof and *ex ante* efficient mechanism guaranteeing positive shares." While they were not able to establish this impossibility, Bogomolnaia et al. showed weaker results, featuring either a much stronger version of strategyproofness or a stronger version of positive share, and additionally requiring symmetry axioms (anonymity and neutrality). Even with the stronger assumptions, their proofs were rather involved, and one of them needed to reason about cases with at least 17 projects. As to whether a distribution rule satisfying the original conditions exists, they left it as "a challenging open question to which we suspect the answer is negative when [the numbers of agents and projects] are large enough."

	UTIL	*CUT*	*NASH*
efficiency	✓	–	✓
fairness	–	✓	✓
strategyproofness	✓	✓	–

Table 1. Standard Axioms

We show that Bogomolnaia et al.'s [2005] conjecture is correct, and the combination of the three axioms leads to an impossibility theorem:

No distribution rule satisfies efficiency, strategyproofness, and positive share when $m \geq 4$ and $n \geq 6$.

We prove this using a computer-aided technique based on SAT solvers that has recently gathered a lot of attention in computational social choice [e.g., Brandl et al., 2018, Geist and Peters, 2017]. We are able to use SAT solvers in our setting since we were able to encode the continuous problem of whether there exists a distribution rule satisfying the three axioms into a combinatorial statement. As a welcome byproduct, our proof goes through with a significantly weaker form of strategyproofness than the one considered by Bogomolnaia et al. [2005]. Using minimal unsatisfiable sets (MUSes), we were then able to automatically construct a proof of the impossibility theorem. This proof is human-readable, though it is very long, and requires to reason about 386 preference profiles. This is certainly the most complicated computer-aided proof in social choice theory (the previous "record holder" uses 47 preference profiles [Brandl et al., 2018]) and perhaps the longest proof of an impossibility theorem in social choice thus far. We have reason to believe that shorter proofs do not exist, and so it is unlikely that the conjecture could have been settled by hand.

The computer surprised us several times during our work on this impossibility. While Bogomolnaia et al. [2005] suspected that the impossibility would only hold for large numbers of agents and projects, we only need $n \geq 6$ agents and $m \geq 4$ projects (and both of these bounds are tight). Moreover, when we add the symmetry axioms used by Bogomolnaia et al. [2005], the SAT solver finds an extremely simple 2-step impossibility proof for $n = 5$ and $m = 4$, which is much more instructive than the existing proof of a weaker result requiring $m \geq 17$. Finally, the SAT approach allowed us to easily check whether the impossibility still holds when slightly weakening the definition of strategyproofness. Using this process, we discovered that any symmetric, efficient, and fair voting rule can be manipulated using the simple strategy of dropping popular projects from one's approval set.

When confronted with an impossibility theorem, the social choice theorist's instinct is to look for "ways out". Table 1 shows that if we drop any of the three axioms (efficiency, strategyproofness, fairness), then the remaining axioms are satisfied by an otherwise very attractive voting rule. Another approach is to try to weaken the axioms. For example, we can weaken efficiency to only require that the outcome distribution is not Pareto dominated by another *fair* distribution. This is a natural definition in settings where fairness is essentially a feasibility requirement. The conditional utilitarian rule satisfies this weakened form of efficiency, making it a strong contender for practical applications. We can also try to weaken strategyproofness. While our computer analysis shows that even very weak forms of the combinatorial variant of strategyproofness give rise to impossibilities, one attractive condition is *monotonicity*, which we show is also a weakening of strategyproofness. It requires that if an agent adds project x to their approval set, then the probability assigned to x must not decrease. Surprisingly, none of the previously known distribution rules is efficient, monotonic, and fair: the utilitarian rule is monotonic but fails positive share, the conditional utilitarian rule is monotonic but fails efficiency, and the Nash rule is efficient and fair but fails monotonicity. The egalitarian rule that maximizes leximin welfare also fails monotonicity. However, we introduce a new rule which we call the *sequential utilitarian rule* which satisfies all three of these properties:

There exists a distribution rule that satisfies efficiency, monotonicity, and positive share.

The sequential utilitarian rule is reasonably natural, though it was specifically designed to satisfy these three axioms, and thus it fails some other desirable properties such as participation (even though the smallest uniform-contribution counterexample we know uses $n \geq 45$ agents, so this deficiency may not be significant in practice). Still, the general approach we took while designing the rule may yield other rules with otherwise unattained axiomatic properties.

2 RELATED WORK

Our paper follows the model introduced and studied by Bogomolnaia et al. [2005], Duddy [2015], and Aziz et al. [2019]. In some parts, we follow Brandl et al. [2020], who noted that distribution rules with variable contributions can also be used for donor coordination, where the budget to be divided is obtained by pooling agents' contributions. These papers can be viewed as studying a type of *participatory budgeting*, which is surveyed by Aziz and Shah [2020]. Other relevant work in this research stream was done by Fain et al. [2016] who allow for linear utilities. They introduce the fairness notion of the *core*, draw a connection between the Nash rule and *Lindahl equilibria* [Foley, 1970], and construct a distribution rule based on differential privacy, which satisfies approximate versions of efficiency, strategyproofness, and the fairness notion of the core. Their rule can be seen as an attempt to avoid the impossibility we present here, but at the cost of severely relaxing each axiom. Airiau et al. [2019] consider fair distribution rules for rankings over projects instead of utility functions. Michorzewski et al. [2020] consider dichotomous preferences and study the *price of fairness* by quantifying what fraction of optimum utilitarian welfare can be achieved by fair distribution rules; Tang et al. [2020] do the same for egalitarian welfare. Freeman et al. [2019] study a related setting where agents report their favorite distribution instead of preferences over projects, and present strategyproof rules that are either efficient or fair.

Participatory budgeting is more commonly studied in a model where projects are indivisible and come with a fixed cost [Aziz and Shah, 2020]. Such projects can either be fully funded or not at all. This model captures elections now run by many city governments [Cabannes, 2004]. The design of voting rules for this model has received much recent attention [e.g., Benade et al., 2017, Fain et al., 2018, Goel et al., 2019] including the case of dichotomous preferences [e.g., Aziz et al., 2018, Talmon and Faliszewski, 2019]. Aziz and Ganguly [2021] have studied a version of the donor coordination setting for indivisible projects.

Our model can also be interpreted as *probabilistic social choice* for the special case of dichotomous preferences [e.g., Brandl et al., 2016, 2018, Brandt, 2017, Gibbard, 1977, Hylland, 1980]. Probabilistic social choice studies functions that map preference profiles to lotteries over alternatives. While this is mathematically equivalent to distributing a fixed budget over public projects, the literature has focused on slightly different axioms due to the different interpretations of the output. For example, depending on the application, probabilistic social choice functions may only be acceptable if they resort to randomization in exceptional cases [e.g., Brandl et al., 2016]. This is in conflict with the idea of fairness, which typically sets lower bounds on the probabilities of alternatives. Our results can be viewed as results in probabilistic social choice by letting all contributions be identical, i.e., each agent brings the same amount of probability mass to the table. The restriction to dichtomous preferences allows for much more positive results than the more general setting, where efficiency is already incompatible with either one of strategyproofness and strict participation (see Remark 1) under mild additional assumptions.

3 MODEL AND AXIOMS

Let A be a finite set of *m projects* and N a finite set of *n agents*. For each $i \in N$, agent i's *contribution* is $C_i \in \mathbb{R}_{>0}$, which we assume to be fixed throughout. The sum of all individual contributions is called the *endowment* $C = \sum_{i=1}^{n} C_i$. Contributions may be monetary contributions to a common pool, or may just be exogenously given voter weights. We often study the case when all individual contributions are equal, and we refer to these as *uniform contributions*. When interpreting distributions as lotteries, for example, the contribution of each agent is $1/n$ of probability mass.

A *distribution* $\delta : A \to \mathbb{R}_{\geq 0}$ with $\sum_{x \in A} \delta(x) = V$ is a function that describes how some amount V is distributed among projects. We write $\Delta(V)$ for the set of all distributions of value V. For convenience, we write distributions as linear combinations of projects, so that $a + 2b$ stands for the distribution δ with $\delta(a) = 1$ and $\delta(b) = 2$ and $\delta(x) = 0$ for all other x. The *support* of a distribution δ is the set of all projects x for which $\delta(x) > 0$.

Agents have preferences over distributions. Following Bogomolnaia et al.'s [2005] seminal work, we study *dichotomous preferences*, where agent $i \in N$ assigns a utility $u_i(x) \in \{0, 1\}$ to each project $x \in A$, and thus assigns utility $u_i(\delta) = \sum_{x \in A} u_i(x) \cdot \delta(x)$ to a distribution δ. We say that agent i *approves* project x if $u_i(x) = 1$ and write $A_i = \{x \in A : u_i(x) = 1\}$ for i's *approval set*. A *profile* $\mathcal{A} = (A_1, \dots, A_n)$ is a tuple of non-empty approval sets, one for each agent.

A *distribution rule* is a function f that maps each profile $\mathcal{A}$ to a distribution $f(\mathcal{A}) = \delta \in \Delta(C)$. We say that f is *neutral* if for every permutation $\sigma : A \to A$ of the alternatives and every profile $\mathcal{A}$, we have $f(\sigma(\mathcal{A})) = \sigma(f(\mathcal{A}))$, where $\sigma(\mathcal{A}) = (\sigma(A_1), \dots, \sigma(A_n))$ is the profile obtained from $\mathcal{A}$ by relabeling alternatives according to σ. We say that f is *anonymous* if its output is invariant under permuting the agents' approval sets and contributions according to the same permutation. All distribution rules that we study in Section 4 are both anonymous and neutral.

There are two distinct interpretations of the model we have now set up. The more common interpretation that motivated the work of Bogomolnaia et al. [2005] and Aziz et al. [2019] views the endowment C as a fixed budget that is provided exogenously. On the other hand, Brandl et al. [2020] were motivated by the problem of donor coordination, where each agent owns the resources C_i which are then pooled into C and then divided (e.g. among charities) via voting. In the latter case, because agents own their contribution, they have a strong claim that their contribution is used in accordance with their wishes, which will be guaranteed by some fairness axioms that we discuss in Section 3.2. Also, in principle, an agent could decide to withdraw from the process and keep C_i for their private use, so we will need rules that provide incentives to contribute resources to the common pool as discussed in Section 3.5. In contrast, with an exogenous budget, the available amount C does not change whether a given agent participates or not, and thus a good distribution rule then merely needs to incentivize submitting a ballot.

3.1 Efficiency

Pareto efficiency is one of the most central properties in microeconomic theory. In our setting, a distribution is efficient if no other distribution yields at least as much utility for every agent and a strictly higher utility for at least one agent. A distribution rule is efficient if it returns an efficient distribution for every profile.

Definition 1 (Efficiency)**.** Let $\mathcal{A}$ be a profile. A distribution $\delta' \in \Delta(C)$ *dominates* another distribution $\delta \in \Delta(C)$ if $u_i(\delta') \geq u_i(\delta)$ for all $i \in N$ and $u_i(\delta') > u_i(\delta)$ for some $i \in N$. δ is *efficient* if no distribution dominates it. A distribution rule f is efficient if $f(\mathcal{A})$ is efficient for all profiles $\mathcal{A}$.

In particular, efficiency implies *ex post* efficiency, which requires that $\delta(x) = 0$ if x is Pareto dominated (that is, there is a project y such that every agent who approves x also approves y and at least one agent approves y but not x). To see this, note that the distribution $\delta' = \delta - \delta(x) \cdot x + \delta(x) \cdot y$ obtained from δ by shifting the probability on x to y dominates δ. The example in Figure 2 shows that efficiency requires more than just *ex post* efficiency: in the profile discussed there, all allocations that give a positive amount of resources to both c and d are dominated because these resources can be efficiently redistributed to a and b. Efficiency is a non-trivial axiom and checking whether a distribution is efficient is typically done via linear programming [e.g., Aziz et al., 2015].

3.2 Fairness

In many contexts, we want the output of a distribution rule to be fair to the agents, which at the minimum should mean that no agent is ignored by the rule. For example, when we vote over the day of the week on which a recurring seminar is to be scheduled, fairness would require that every participant should be able to attend at least some of the seminars. If the schedule alternates between days according to a distribution rule, this fairness requirement can easily be accommodated. We formalize this idea in three axioms taken from the existing literature [Aziz et al., 2019, Bogomolnaia et al., 2002, 2005, Duddy, 2015] and adapted to our setting which does not require uniform contributions.

Definition 2 (Fair share)**.** Given a profile $\mathcal{A}$, a distribution $\delta \in \Delta(C)$ satisfies

- *positive share* if for all $i \in N$, $u_i(\delta) > 0$,
- *individual fair share* if for all $i \in N$, $u_i(\delta) \geq C_i$,
- *group fair share* if for all sets $S \subseteq N$, $\sum_{x \in \bigcup_{i \in S} A_i} \delta(x) \geq \sum_{i \in S} C_i$.

A distribution rule f satisfies positive share (resp. individual fair share or group fair share) if $f(\mathcal{A})$ satisfies the respective property for all profiles $\mathcal{A}$.

Positive share is extremely weak and only rules out situations where an agent is not served at all. Individual fair share requires that not only a positive amount but at least an amount equal to an agent's contribution C_i should be spent on projects approved by that agent. Group fair share (which is sometimes also called proportional sharing) additionally gives guarantees to all groups of agents. It demands that at least an amount equal to the total contributions of the group should be spent on projects approved by some group member.

In the donor coordination interpretation of our model, the endowment C is obtained by pooling resources C_i that agents own. Thus, each agent i has a justified claim that their contribution C_i is only spent on projects in A_i approved by i. On first sight, individual fair share provides such a guarantee, but it is actually too weak. Consider three agents with uniform contributions and

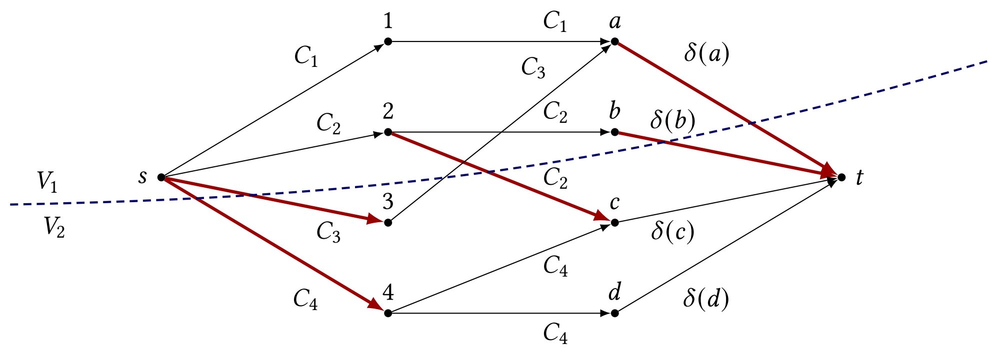

Fig. 1. Illustration of the proof of Proposition 1. In this example, $A_1 = \{a\}$, $A_2 = \{b, c\}$, $A_3 = \{a\}$, and $A_4 = \{c, d\}$. The dashed line indicates a cut (V_1, V_2). Hence, $N_1' = \{1\}$, $N_1'' = \{2\}$, $N_2 = \{3, 4\}$, $B_1 = \{a, b\}$, and $B_2 = \{c, d\}$. Edges that start in V_1 and end in V_2 count toward the value of the cut and are drawn thicker.

the profile $\mathcal{A} = (\{a\}, \{a\}, \{b\})$. Then the distribution $\delta = 1a + 2b$ satisfies individual fair share, yet we cannot tell both agents 1 and 2 that their contribution ($C_1 + C_2 = 2$) was spent on their approved project a. This motivates a different fairness notion, introduced by Brandl et al. [2020], called decomposability. It requires that the distribution δ can be expressed as a sum of individual distributions $\delta_i \in \Delta(C_i)$ that only spend on projects approved by i, so $\delta_i(x) = 0$ for all $x \in A \setminus A_i$.

Definition 3 (Decomposability)**.** Let $\mathcal{A}$ be a profile. A distribution $\delta \in \Delta(C)$ is *decomposable* if we can write $\delta = \sum_{i \in N} \delta_i$ where for each $i \in N$, $\delta_i \in \Delta(C_i)$ is an *individual distribution* satisfying $u_i(\delta_i) = C_i$. A distribution rule f is decomposable if $f(\mathcal{A})$ is decomposable for all profiles $\mathcal{A}$.

An advantage of such a decomposition is that it can be used to explain to the agents what their contributions have been used for. In some cases, decomposability is almost indispensable, such as in applications where the individual contributions are not collected from the agents, but the rule merely advises each agent how to spend their resources. This is, for example, the case in a decentralized version of the donor coordination setting discussed by Brandl et al. [2020].

Interestingly, it turns out that decomposability and group fair share are equivalent. This result generalizes Hall's theorem for bipartite matching.[1]

Proposition 1. A distribution $\delta \in \Delta(C)$ is decomposable if and only if it satisfies group fair share.

Proof. Suppose δ is decomposable, with $\delta = \sum_{i \in N} \delta_i$. Then for any $S \subseteq N$, we have $\sum_{x \in \bigcup_{i \in S} A_i} \delta(x) \geq \sum_{i \in S} \sum_{x \in A_i} \delta_i(x) \geq \sum_{i \in S} C_i$. Thus, δ satisfies group fair share.

We prove the converse direction by an application of the max-flow min-cut theorem. We start by setting up a flow network, i.e., a weighted digraph (V, E, c), where V is a set of vertices, E is a set of (directed) edges, and c is a function that assigns to each edge a capacity. Set $V = N \cup A \cup \{s, t\}$, where s and t are the source and the sink, respectively. For each $i \in N$, add an edge with capacity C_i from s to i and, for each $x \in A_i$, an edge with capacity C_i from i to x. For each $x \in A$, add an edge with capacity $\delta(x)$ from x to t.

A flow in this network with value $C = \sum_{i \in N} C_i$ provides a decomposition of $\delta = \sum_{i \in N} \delta_i$, where $\delta_i(x)$ is the flow along the edge from i to x. We prove that there exists such a flow by showing that every cut has value at least C. The proof is illustrated in Figure 1.

[1]A similar result for the case when distributions and contributions are integral has been shown in a different context by Bokal et al. [2012, Theorem 20].

A cut is a partition (V_1, V_2) of V such that $s \in V_1$ and $t \in V_2$. The value $v(V_1, V_2)$ of a cut is the sum of the capacities of edges from V_1 to V_2, i.e., $v(V_1, V_2) = \sum_{e \in (V_1 \times V_2) \cap E} c(e)$. Given a cut (V_1, V_2), let $N_j = N \cap V_j$ and $B_j = A \cap V_j$ for $j = 1, 2$. We decompose N_1 further into $N_1' = \{i \in N_1 : A_i \cap B_2 = \emptyset\}$ and $N_1'' = N_1 \setminus N_1'$. Notice that, by assumption, $\sum_{x \in B_1} \delta(x) \geq \sum_{i \in N_1'} C_i$ since no agent in N_1' approves a project in B_2. We can see that

$$v(V_1, V_2) = \sum_{x \in B_1} \delta(x) + \sum_{i \in N_1''} \sum_{x \in A_i \cap B_2} C_i + \sum_{i \in N_2} C_i \geq \sum_{i \in N_1'} C_i + \sum_{i \in N_1''} C_i + \sum_{i \in N_2} C_i = C$$

The first sum counts edges from B_1 to t, the double sum counts edges from N_1 to B_2 (all of which start in N_1''), and the third sum counts edges from s to N_2. For the inequality we use that $\sum_{x \in B_1} \delta(x) \geq \sum_{i \in N_1'} C_i$ as noted above and the fact that $|A_i \cap B_2| \geq 1$ for $i \in N_1''$. Thus, every cut has value at least C, which implies that there is a flow of value C. □

Decomposability is a natural condition, and in some contexts only decomposable distributions may be acceptable. In such cases, our definition of efficiency is needlessly strong, because it would not be relevant if a distribution is only dominated by non-decomposable distributions. Thus, we can consider the following weakening of efficiency.

Definition 4 (Decomposable efficiency)**.** A distribution rule f satisfies *decomposable efficiency* if for all profiles $\mathcal{A}$, the distribution $f(\mathcal{A})$ is not dominated by a decomposable distribution.

3.3 Strategyproofness

In the remainder of this section, we investigate the agents' incentives and a monotonicity property. In our model, an agent submits a set of approved projects to the distribution rule. Strategyproofness requires that an agent's utility when reporting their approval set truthfully is at least as large as for any other report. Equivalently, whenever an agent deviates from truthful reporting, the amount of the endowment assigned to their (truthfully) approved projects weakly decreases.

Definition 5 (Strategyproofness)**.** A distribution rule f is *strategyproof* if for all $i \in N$ and all profiles $\mathcal{A}$ and $\mathcal{A}'$ with $A_j = A_j'$ for all $j \neq i$, we have $u_i(f(\mathcal{A})) \geq u_i(f(\mathcal{A}'))$.

In many mechanism design settings without transferable utility, only degenerate mechanisms that ignore most of the preference information, such as dictatorships and combinations thereof, are strategyproof [Gibbard, 1977, Hylland, 1980]. But dichotomous preferences allow for more positive results. For example, in social choice, Brams and Fishburn [1983] showed that approval voting is strategyproof under dichotomous preferences. In our setting, there also exist attractive strategyproof distribution rules. In particular, the utilitarian and the conditional utilitarian rules (see Section 4) are strategyproof [Aziz et al., 2019, Bogomolnaia et al., 2005].

3.4 Monotonicity

Monotonicity requires that if a project becomes more popular among the agents, the amount of the endowment allocated to it must not decrease. This property also appears as non-perverseness in the work of Gibbard [1977] on probabilistic social choice.

Definition 6 (Monotonicity)**.** A distribution rule f is *monotonic* if for any profiles $\mathcal{A}$ and $\mathcal{A}'$, any agent $i \in N$, and any project $x \notin A_i$, if $A_i' = A_i \cup \{x\}$ and $A_j' = A_j$ for all $j \neq i$, we have $f(\mathcal{A}')(x) \geq f(\mathcal{A})(x)$.

Monotonicity can be interpreted as an incentive property from the point of view of those in charge of a particular project: without monotonicity, a project manager may want to dissuade agents from voting for their project. In addition, as we show next, monotonicity is implied by strategyproofness in our model.[2] Thus monotonicity is also an incentive property for agents.

Proposition 2. Every strategyproof distribution rule is monotonic.

Proof. Let $\mathcal{A}$ and $\mathcal{A}'$ be two profiles so that for some agent $i \in N$ and some project $x \notin A_i$, we have $A'_i = A_i \cup \{x\}$ and $A'_j = A_j$ for all $j \neq i$. Let $\delta = f(\mathcal{A})$ and $\delta' = f(\mathcal{A}')$. Strategyproofness implies that in profile $\mathcal{A}$, agent i does not want to change her report from A_i to A'_i, so we must have $\sum_{a \in A_i} \delta'(a) \leq \sum_{a \in A_i} \delta(a)$. Similarly, in profile $\mathcal{A}'$, agent i does not want to change her report from A'_i to A_i, so we have $\sum_{a \in A_i} \delta(a) + \delta(x) \leq \sum_{a \in A_i} \delta'(a) + \delta'(x)$. Adding the two inequalities and cancelling identical terms gives $\delta(x) \leq \delta'(x)$, as required. □

3.5 Contribution Incentives

As we mentioned above, in the donor coordination interpretation of our model, agents need to be incentivized to contribute their resources to the mechanism. Under this interpretation, agents own their resources and could presumably use them productively outside the mechanism (e.g., they could send their money directly to a charity rather than go through the pooling procedure). Because pooling can lead to efficiency gains, we want to encourage agents to contribute to the pool.

Brandl et al. [2020] proposed a formal property that captures this idea, which they call *contribution incentive-compatibility*. Their main result is that the Nash rule satisfies it (even without assuming dichotomous preferences), and they argue that this makes the Nash rule well-suited for donor coordination since no other efficient rule is known to incentivize contributions.

To formally state Brandl et al.'s [2020] axiom (restricted to dichotomous preferences), we need to augment our model by allowing for varying sets of agents. Use the natural numbers $\mathbb{N}$ to index a universe of agents and denote by $\mathcal{F}(\mathbb{N})$ the collection of finite and non-empty subsets of $\mathbb{N}$. The endowment of the agents in $N \in \mathcal{F}(\mathbb{N})$ is $C_N = \sum_{i \in N} C_i$. A profile for the set of agents $N \in \mathcal{F}(\mathbb{N})$ is a tuple of approval sets $\mathcal{A} = (A_i)_{i \in N}$. If $|N| \geq 2$ and $j \in N$, we write $A_{-j} = (A_i)_{i \in N \setminus \{j\}}$ for the profile obtained by removing agent j. A distribution rule is now a function f that maps any profile $\mathcal{A}$ defined on any set of agents $N \in \mathcal{F}(\mathbb{N})$ to a distribution $f(\mathcal{A}) = \delta \in \Delta(C_N)$.

We assume that even if an agent does not participate in the distribution rule, the agent nevertheless receives utility from the other agents' contributions to the projects. This assumption holds because projects are public goods. Thus an agent i can choose between two options: either contribute C_i to the endowment C_N and benefit from the money that $f(\mathcal{A})$ assigns to projects approved by i, or else keep C_i and benefit from it and additionally benefit from the money that $f(\mathcal{A}_{-i})$ happens to assign to approved projects. Strong contribution incentive-compatibility requires that the former choice is at least as good as the latter for i.

Definition 7 (Contribution incentive-compatibility)**.** A distribution rule f satisfies *contribution incentive-compatibility* if for every profile $\mathcal{A}$ for agents $N \in \mathcal{F}(\mathbb{N})$ with $|N| \geq 2$ and every agent $i \in N$, $u_i(f(\mathcal{A})) \geq u_i(f(\mathcal{A}_{-i})) + C_i$.

This definition assumes that the "outside option" of keeping one's contribution has the same utility as an approved project. An assumption at the opposite end of the spectrum would be that one's contribution has no value outside the mechanism. In that case, a much weaker property suffices to incentivize contribution.

Definition 8 (Weak participation)**.** A distribution rule f satisfies *weak participation* if for all profiles $\mathcal{A}$ for agents $N \in \mathcal{F}(\mathbb{N})$ with $|N| \geq 2$ and every agent $i \in N$, we have $u_i(f(\mathcal{A})) \geq u_i(f(\mathcal{A}_{-i}))$.

[2] Similarly, Gibbard [1977] shows that his notion of strategyproofness implies non-perverseness. This result is not formally related to ours because of differences in the model.

Utilitarian

	a	b	c	d
δ_1	1	·	·	·
δ_2	1	·	·	·
δ_3	1	·	·	·
δ_4	1	·	·	·
δ_5	1	·	·	·
δ_6	1	·	·	·
$\sum$	6	·	·	·

Conditional Utilitarian

	a	b	c	d
δ_1	1	·	·	·
δ_2	1	·	·	·
δ_3	·	0.5	0.5	·
δ_4	·	0.5	·	0.5
δ_5	1	·	·	·
δ_6	1	·	·	·
$\sum$	4	1	0.5	0.5

Nash Product

	a	b	c	d
δ_1	1	·	·	·
δ_2	1	·	·	·
δ_3	·	1	·	·
δ_4	·	1	·	·
δ_5	1	·	·	·
δ_6	0.6	0.4	·	·
$\sum$	3.6	2.4	·	·

Sequential Utilitarian

	a	b	c	d
δ_1	1	·	·	·
δ_2	1	·	·	·
δ_3	·	1	·	·
δ_4	·	1	·	·
δ_5	1	·	·	·
δ_6	1	·	·	·
$\sum$	4	2	·	·

Fig. 2. Example outcomes of distribution rules on profile $\mathcal{A} = (\{a, c\}, \{a, d\}, \{b, c\}, \{b, d\}, \{a\}, \{a, b, c, d\})$ with uniform contributions. The last row of each table shows the distribution returned by the distribution rule. The other rows show a division of this distribution into individual distributions (zeros omitted). Approved projects are highlighted in grey.

Obviously, contribution incentive-compatibility implies weak participation. It also implies individual fair share, but is logically independent of decomposability [see Brandl et al., 2020].

Remark 1. Based on earlier work by Moulin [1988b] and Brandl et al. [2015], Aziz et al. [2019] consider two axioms they call *strict participation* and *participation* and which are appropriate when the size of the endowment is independent of the set of participating agents (e.g. because the budget is given exogenously rather than being obtained from pooling agents' contributions). In our language, the participation axiom requires that the *fraction* of the endowment spent on projects approved by agent i decreases weakly if agent i abstains (that is, $\frac{1}{C}u_i(f(\mathcal{A})) \geq \frac{1}{C-C_i}u_i(f(\mathcal{A}_{-i}))$). Strict participation requires a strict inequality except when $u_i(f(\mathcal{A}_{-i})) = C - C_i$. One can check that contribution incentive-compatibility implies strict participation, that strict participation implies participation, and that participation implies weak participation. All of these implications are strict.

4 RULES

We now define the four distribution rules that we study in this paper. The outcomes of these rules for an example profile are illustrated in Figure 2. All rules can be efficiently computed (computing the *NASH* distribution *exactly* is not possible since it may be irrational, but it can be approximated efficiently using convex programming).

For a set $A' \subseteq A$, we write $\text{uni}(A') \in \Delta(1)$ for the distribution δ with $\delta(x) = 1/|A'|$ for all $x \in A'$, and $\delta(y) = 0$ for all $y \in A \setminus A'$.

4.1 Utilitarian Rule

The standard way of attaining efficiency in mechanism design is by maximizing a notion of social welfare. The literature has identified three central versions of this concept: utilitarian welfare, egalitarian welfare, and the Nash product [Moulin, 1988a]. The four distribution rules that follow are all in some way based on optimizing welfare.[3]

The simplest rule following this recipe is the *utilitarian rule*. It returns a distribution δ which maximizes the contribution-weighted sum of agents' utilities $\sum_{i \in N} C_i \cdot u_i(\delta)$.[4] These are exactly those distributions in which the endowment is distributed only on the projects with the highest

[3]Aziz et al. [2019] study a rule called *EGAL* based on maximizing egalitarian/leximin welfare, but we will not discuss it here, since each of the axioms from Section 3 that is satisfied by *EGAL* is also satisfied by *NASH*.

[4]This is a variant of the traditional utilitarian rule because agents are weighted by the size of their contribution. This variant is more robust, and is continuous in the contributions C_i. Without weighting, agents with extremely small individual

score, where the *score* $n_x = \sum_{i\in N: x\in A_i} C_i$ of a project x is the combined contribution of all agents who approve it. To see this, note that if any part of the endowment is spent on other projects, utilitarian welfare can be increased by redistributing it to a project with a higher score. Note that there might be several projects that have the same score, and so there may be many distributions maximizing utilitarian welfare. If so, for concreteness, we let the endowment be distributed uniformly among them. Let $A^{\max} = \{x \in A \colon n_x \geq n_y \text{ for all } y \in A\}$ be the set of projects with the highest score. Then,

$$UTIL(\mathcal{A}) = C \cdot \text{uni}(A^{\max}). \qquad \text{(Utilitarian Rule)}$$

We now check which axioms *UTIL* satisfies.

✓**efficiency** is satisfied because any distribution dominating $UTIL(\mathcal{A})$ would have strictly higher utilitarian welfare.

✗ **positive share** is failed. For example, see Figure 2, or consider the profile $\mathcal{A} = (\{a\}, \{a\}, \{b\})$ with uniform contributions where $UTIL(\mathcal{A}) = 3a$, violating positive share for agent 3. It follows that *UTIL* also fails individual fair share and decomposability (group fair share).

✓**strategyproofness** is satisfied [Bogomolnaia et al., 2005], for the same reason that approval voting is strategyproof under dichotomous preferences.

✓**monotonicity** is satisfied because strategyproofness implies monotonicity (Proposition 2).

✗ **contribution incentive-compatibility** is failed since it is stronger than positive share. However, *UTIL* satisfies participation [Aziz et al., 2019].

4.2 Conditional Utilitarian Rule

A natural way to obtain a fairer distribution rule while keeping the spirit of utilitarian welfare is to select a distribution that maximizes welfare among decomposable distributions only. This is what the *conditional utilitarian rule* does. It was first considered by Duddy [2015] and further analyzed by Aziz et al. [2019]. Like for *UTIL*, it is possible to explicitly describe the solutions of this constrained optimization problem: each agent needs to distribute her contribution among those of her approved projects which have highest score. Again, for concreteness, we let agents split uniformly in the event of ties. Formally, we write $A_i^{\max} = \{x \in A_i \colon n_x \geq n_y \text{ for all } y \in A_i\}$ for the projects with the highest score within the approval set of agent i. Then,

$$CUT(\mathcal{A}) = \sum_{i\in N} C_i \cdot \text{uni}(A_i^{\max}). \qquad \text{(Conditional Utilitarian Rule)}$$

By design, this rule is decomposable and maximizes utilitarian welfare among all decomposable distributions. With Proposition 1, we can also describe *CUT* as the rule that maximizes utilitarian welfare subject to satisfying group fair share.

✗ **efficiency** is failed as shown in the example in Figure 2. All distributions that put a positive amount of resources on c and d are dominated (see Section 3.1). However, it is clear from its definition that *CUT* satisfies decomposable efficiency (Definition 4).

✓**decomposability** is satisfied by construction.

✓**strategyproofness** is satisfied [Aziz et al., 2019].

✓**monotonicity** is satisfied because strategyproofness implies monotonicity (Proposition 2).

✓**contribution incentive-compatibility** is satisfied as we show next in Proposition 3.

Proposition 3. *CUT* is contribution incentive-compatible.

contributions would get the same influence as agents with large individual contributions and each agent would be incentivized to “split up” into several agents with the same preferences.

Proof. Let $\mathcal{A}$ be a profile and let $i \in N$. Write $\delta = CUT(\mathcal{A})$ and $\tilde{\delta} = CUT(\mathcal{A}_{-i})$, and consider the sets $A_j^{\max}$ and $\tilde{A}_j^{\max}$ associated with $\mathcal{A}$ and $\mathcal{A}_{-i}$ respectively. We claim that for each $j \in N \setminus \{i\}$,

$$\text{either } A_j^{\max} \subseteq A_i \text{ or } \tilde{A}_j^{\max} \cap A_i = \emptyset.$$

If not, then there exists $x \in A_j^{\max} \setminus A_i$ and $y \in \tilde{A}_j^{\max} \cap A_i$. Since $x, y \in A_j$, by the definition of $A_j^{\max}$ and $\tilde{A}_j^{\max}$, we have $n_x \geq n_y$ and $\tilde{n}_y \geq \tilde{n}_x$. Hence, we have

$$n_y = \tilde{n}_y + C_i \geq \tilde{n}_x + C_i > \tilde{n}_x = n_x \geq n_y,$$

which is a contradiction. It follows that

$$u_i(\delta) = C_i + \sum_{j \in N \setminus \{i\}} C_j \frac{|A_i \cap A_j^{\max}|}{|A_j^{\max}|} \geq C_i + \sum_{j \in N \setminus \{i\}} C_j \frac{|A_i \cap \tilde{A}_j^{\max}|}{|\tilde{A}_j^{\max}|} = C_i + u_i(\tilde{\delta}). \qquad \square$$

4.3 Nash Rule

The Nash product, which refers to the product of agent utilities, is often seen as a compromise between utilitarian and egalitarian welfare [Moulin, 1988a]. Maximizing the Nash product has been found to yield "fair" or "proportional" outcomes in many preference aggregation settings, and it also turns out to be attractive in our context. Formally, it is defined as follows.

$$NASH(\mathcal{A}) = \operatorname*{arg\,max}_{\delta \in \Delta(C)} \prod_{i \in N} u_i(\delta)^{C_i} = \operatorname*{arg\,max}_{\delta \in \Delta(C)} \sum_{i \in N} C_i \log u_i(\delta). \qquad \text{(Nash Product Rule)}$$

Just like *UTIL*, this rule is efficient. But, remarkably, it is not necessary to define a "conditional Nash rule": it follows from the results of Guerdjikova and Nehring [2014] that the optimum for the Nash product is always decomposable.[5]

✓**efficiency** is satisfied because a dominating distribution would have strictly higher Nash product.

✓**decomposability** is satisfied, as can be shown by analyzing first-order conditions of optimality [Brandl et al., 2020, Guerdjikova and Nehring, 2014].

✗ **strategyproofness** is failed as a consequence of our impossibility, Theorem 1.

✗ **monotonicity** is failed as shown in Proposition 4 below.

✓**contribution incentive-compatibility** is satisfied, but surprisingly difficult to show [Brandl et al., 2020].

Proposition 4. *NASH* fails monotonicity.

Proof. Consider the 6-agent profile $\mathcal{A} = (\{a\}, \{a, b\}, \{a, c\}, \{b, c, d\}, \{b, d\}, \{c, d\})$ and contributions $C_1 = C_2 = C_3 = 1$ and $C_4 = C_5 = C_6 = 2$ (the same example can be adapted to uniform contributions by doubling the last three agents, resulting in a 9-agent profile). We have $NASH(\mathcal{A}) = 3a + 6d$. If we consider the profile $\mathcal{A}'$ where agent 1 additionally approves project d, i.e., $A_1' = \{a, d\}$, we get $NASH(\mathcal{A}') \approx 1.54a + 0.77b + 0.77c + 5.92d$, with a decreased amount on d. The exact distribution $NASH(\mathcal{A}')$ is $2\kappa a + \kappa b + \kappa c + (9 - 4\kappa)d$ where $\kappa = (7 - \sqrt{22})/3$. We found this example by exhaustive search; no examples with uniform contributions exist for $m = 4$ and $n < 9$. $\square$

[5]In addition, Guerdjikova and Nehring [2014] show that *NASH* satisfies a remarkable fixed point property: let δ be a distribution returned by *NASH*. For each voter $i \in N$, define $\delta_i(x) = \delta(x)/u_i(\delta)$ for all $x \in A_i$. Then $\delta = \sum_{i \in N} \delta_i$. Thus, individual distributions can be retrieved from the total distribution by restricting and rescaling to the corresponding approval set.

Algorithm 1 Sequential Utilitarian Rule

$w^1 \leftarrow e_N$
$s^* \leftarrow \max_{x\in A} s_{w^1}(x)$
$A^1 \leftarrow \arg\max_{x\in A} s_{w^1}(x)$
$N^1 \leftarrow \{i \in N : A_i \cap A^1 \neq \emptyset\}$
$\delta^1 \leftarrow \sum_{i\in N^1} C_i \cdot \text{uni}(A_i \cap A^1)$
for round $k = 2, \ldots, n$ **do**
  $t^* \leftarrow \max\{t \in \mathbb{R} : \max_{x\in A} s_{w^{k-1}+te_{N\setminus N^{k-1}}}(x) \leq s^*\}$
  $w^k \leftarrow w^{k-1} + t^* e_{N\setminus N^{k-1}}$
  $A^k \leftarrow \arg\max_{x\in A} s_{w^k}(x)$
  $N^k \leftarrow \{i \in N : A_i \cap A^k \neq \emptyset\}$
  for $i \in N^k \setminus N^{k-1}$ **do**
    $\delta_i \leftarrow C_i \cdot \text{uni}(A_i \cap A^k)$
  $\delta^k \leftarrow \sum_{i\in N^k\setminus N^{k-1}} \delta_i$
return $\sum_{k=1}^n \delta^k$

4.4 Sequential Utilitarian Rule

Lastly, we will introduce a new distribution rule. To the best of our knowledge, this is the first known distribution rule that satisfies efficiency, monotonicity, and positive share (it is even decomposable). This rule shows that in our main impossibility result (Theorem 2), we cannot weaken strategyproofness to monotonicity.

This rule, which we call the *sequential utilitarian rule* (*SUT*), will decide sequentially for each agent $i \in N$ where to direct i's contribution C_i. We ensure that the rule is efficient by constructing positive weights $w = (w_i)_{i\in N}$ for each agent such that we maintain the invariant that the overall distribution δ under construction maximizes weighted utilitarian welfare $\sum_{i\in N} w_i C_i u_i(\delta)$. Clearly, a distribution maximizing this value is efficient, provided that $w_i > 0$ for all $i \in N$.[6]

On a high level, *SUT* operates as follows. We start with unit weights, $w = (1, \ldots, 1)$, and identify those projects x with maximum w-score $s_w(x) = \sum_{i\in N} w_i C_i u_i(x)$; we call these projects w-maximum. For every agent i who approves a w-maximum project, we split i's individual distribution $\delta_i \in \Delta(C_i)$ uniformly among the w-maximum projects that i approves, and freeze δ_i and w_i. For all other agents, we then raise their weights at a common rate until some additional projects become w-maximum, and set the individual distributions δ_i of not yet frozen agents who now approve a w-maximum project, and freeze δ_i and w_i. We then go back to raising weights, until all individual distributions have been set and frozen. Then

$$SUT(\mathcal{A}) = \sum_{i=1}^{n} \delta_i. \qquad \text{(Sequential Utilitarian Rule)}$$

This procedure is formalized in Algorithm 1. In the pseudocode, for each $S \subset N$, we denote by $e_S \in \mathbb{R}^N$ the vector with 1's in the S coordinates and 0's in the remaining coordinates. At the end of each round $k = 1, \ldots, n$, w^k denotes the positive weights assigned to agents at the end of the round, which was obtained by increasing the weights of not yet frozen agents by t^*. The set A^k is the set of w^k-maximum projects. The set N^k contains the agents who are frozen at the end of round k, and δ^k are the combined individual distributions of the agents frozen in round k. Note that in case an agent i approves multiple w^k-maximum projects, we split C_i uniformly among them.

[6]In fact, using Farkas' lemma, one can show that a distribution is efficient if and only if it maximizes w-weighted utilitarian welfare for some w. This is also known as the *efficiency welfare theorem* [see, e.g., Carroll, 2010].

✓**efficiency** is satisfied because a dominating distribution would have strictly higher w-weighted utilitarian welfare.

✓**decomposability** is satisfied by construction.

✗ **strategyproofness** is failed as a consequence of our impossibility, Theorem 1.

✓**monotonicity** is satisfied as we prove in Proposition 5 below.

✗ **contribution incentive-compatibility** is failed by the example in Proposition 6. The same example shows that even weak participation is failed.

Proposition 5. *SUT* satisfies monotonicity.

Proof. Let $\mathcal{A}$ and $\widehat{\mathcal{A}}$ be identical profiles except that $\hat{A}_i = A_i \cup \{x\}$ for some $x \notin A_i$. We consider how *SUT* (i.e., Algorithm 1) proceeds for both profiles. First, suppose $\hat{s}_{e_N}(x) > \arg\max_{a\in A} s_{e_N}(a) = s^*$. Since $\mathcal{A}$ and $\widehat{\mathcal{A}}$ only differ in that $u_i(x) = 0$ and $\hat{u}_i(x) = 1$, it follows that

$$\hat{s}^* = \hat{s}_{e_N}(x) > s^* \geq s_{e_N}(a) = \hat{s}_{e_N}(a)$$

for all $a \neq x$. Thus, x is the unique project that maximizes the approval score for the profile $\widehat{\mathcal{A}}$, so

$$SUT(\widehat{\mathcal{A}})(x) = \sum_{j\in N\colon x\in\hat{A}_j} C_j = \sum_{j\in N\colon x\in A_j} C_j + C_i > \sum_{j\in N\colon x\in A_j} C_j \geq SUT(\mathcal{A})(x).$$

Thus, monotonicity is satisfied.

Now suppose $\hat{s}_{e_N}(x) \leq \hat{s}^* = s^*$. If $w^k = \hat{w}^k$ for all rounds k, then $s_{w^k}(x) \leq \hat{s}_{\hat{w}^k}(x)$ and $s_{w^k}(a) = \hat{s}_{\hat{w}^k}(a)$ for all $a \in A$ and all k. Consider any agent $j \neq i$. If for the profile $\mathcal{A}$ agent j distributes their contribution uniformly over the set of projects A', then for the profile $\widehat{\mathcal{A}}$, agent j distributes their contribution uniformly over either A' or $A' \cup \{x\}$. In either case, agent j's contribution to x does not decrease from $\mathcal{A}$ to $\widehat{\mathcal{A}}$. Moreover, since $x \notin A_i$, agent i does not distribute anything over x for $\mathcal{A}$. Hence, $SUT(\widehat{\mathcal{A}})(x) \geq SUT(\mathcal{A})(x)$ and monotonicity holds.

Otherwise, let $k \geq 2$ be the index of the first round for which $w^k \neq \hat{w}^k$. (Note that $w^1 = \hat{w}^1$ since we assumed $s^* = \hat{s}^*$.) Then $x \notin A^{k-1}$. For if $x \in A^{k-1}$, we have $s_{w^{k-1}}(x) = s^*$ and hence

$$\hat{s}_{\hat{w}^{k-1}}(x) = s_{w^{k-1}}(x) + w_i^{k-1} \cdot C_i = s^* + w_i^{k-1} \cdot C_i > s^* = \hat{s}^*,$$

which is a contradiction. On the other hand, since $w^k \neq \hat{w}^k$, we have $x \in \hat{A}^k$. Moreover, for $a \neq x$ and $l < k$, $a \in A^l$ if and only if $a \in \hat{A}^l$ since $w^l = \hat{w}^l$ and all agents have the same utilities for a in $\mathcal{A}$ and $\widehat{\mathcal{A}}$. Say that $l \leq k$ is such that $x \in \hat{A}^l \setminus \hat{A}^{l-1}$.

If $l < k$, then $\hat{N}^l = N^l \cup \{j \in N \setminus N^l \colon x \in A_j\} \cup \{i\}$, so that

$$SUT(\widehat{\mathcal{A}})(x) = \sum_{j\in\hat{N}^l\setminus\hat{N}^{l-1}\colon x\in\hat{A}_j} \frac{C_j}{|\hat{A}_j \cap \hat{A}^l|} \geq \sum_{j\in N\setminus N^l\colon x\in A_j} C_j \geq SUT(\mathcal{A})(x),$$

where we used $A_j \cap A^l = \emptyset$ if $j \notin N^l$ for the first inequality.

If $l = k$, i.e., $x \in \hat{A}^k \setminus \hat{A}^{k-1}$, then $\hat{N}^k = N^k \cup \{j \in N \setminus N^k \colon x \in A_j\} \cup \{i\}$, so that

$$\begin{aligned} SUT(\widehat{\mathcal{A}})(x) &= \sum_{j\in\hat{N}^k\colon x\in\hat{A}_j} \frac{C_j}{|\hat{A}_j \cap \hat{A}^k|} \\ &\geq \sum_{j\in N\setminus N^k\colon x\in A_j} C_j + \sum_{j\in N^k\setminus N^{k-1}\colon x\in A_j} \frac{C_j}{|A_j \cap \hat{A}^k|} + \frac{C_i}{|\hat{A}_i \cap \hat{A}^k|} \geq SUT(\mathcal{A})(x). \end{aligned}$$

Hence, monotonicity is satisfied. □

Initially, we hoped that *SUT* also satisfies contribution incentive-compatibility because an exhaustive search among profiles with uniform contributions did not yield a counter-example for $m = 4$ and $n \leq 14$ nor for $m = 5$ and $n \leq 10$. However, we then manually constructed the counter-example below. It can be transformed into a uniform contribution counter-example with 45 agents.

Proposition 6. *SUT* fails weak participation.

Proof. Consider 8 agents with $C_1 = 6$, $C_2 = 9$, $C_3 = C_4 = 6$, $C_5 = 12$, $C_6 = 3$, $C_7 = 2$, and $C_8 = 1$, for a total endowment of $C = 45$. Let $\mathcal{A} = (\{a,b,c\}, \{a\}, \{b,d\}, \{c,d\}, \{a,b,c,e\}, \{d,e\}, \{e\}, \{a,e\})$. Then one can calculate that $SUT(\mathcal{A}) = 28a+6b+6c+3d+2e$ and $SUT(\mathcal{A}_{-8}) = 27a+6b+6c+1.5d+3.5e$. But then $u_8(SUT(\mathcal{A})) = 30 < 30.5 = u_8(SUT(\mathcal{A}_{-8}))$, and so weak participation is violated. □

5 IMPOSSIBILITY

In this section, we present our main result showing that no distribution rule satisfies efficiency, strategyproofness, and positive share, as well as the method we used to obtain this result.

5.1 An Easy Impossibility Theorem

To warm up, we first prove a weaker impossibility theorem which additionally uses the anonymity and neutrality axioms. The proof is remarkably simple, and only reasons about the behavior of the rule on two preference profiles. Indeed, one can view it as a *universal counterexample*: it identifies two specific profiles which admit a manipulation for all efficient rules satisfying positive share.

Theorem 1. No anonymous and neutral distribution rule satisfies efficiency, strategyproofness, and positive share when $m \geq 4$ and $n \geq 5$.

Proof. We prove the incompatibility for $m = 4$ and $n = 5$. The proof can be adapted to larger values by adding agents approving all projects or by adding projects which no-one approves.

Assume there is a strategyproof distribution rule f satisfying efficiency and positive share. Now consider a profile $\mathcal{A}$ with uniform contributions $C_i = 1$ for all agents $i \in N$ and the profile

$$\mathcal{A} = (\{a,c\}, \{a,d\}, \{b,c\}, \{a,b\}, \{a\}) .$$

Let $\delta = f(\mathcal{A})$ be the distribution returned by the distribution rule. Because f is anonymous and neutral, since b and c are symmetric, we must have $\delta(b) = \delta(c)$, and this value must be positive by positive share for agent 3. It follows that $u_4(\delta) < C$ because a positive amount is spent on project c, which agent 4 does not approve.

Suppose agent 4, who approves $\{a,b\}$, instead reports $\{b,d\}$. The resulting profile is

$$\mathcal{A}' = (\{a,c\}, \{a,d\}, \{b,c\}, \{b,d\}, \{a\}) .$$

Let $\delta' = f(\mathcal{A}')$ be the distribution now returned by the distribution rule. Now c and d are symmetric projects in $\mathcal{A}'$, and thus we must have $\delta'(c) = \delta'(d)$ by anonymity and neutrality of f. If $\delta'(c) = \delta'(d)$ is positive, say equal to $\epsilon > 0$, then δ' is Pareto dominated by the distribution obtained from δ' by moving ϵ from c to a and ϵ from d to b. This contradicts efficiency of f. Thus $\delta'(c) = \delta'(d) = 0$ and the entire endowment is distributed between projects a and b, and so $u_4(\delta') = C$, where we take agent 4's utility as reported in profile $\mathcal{A}$, and in particular $u_4(\delta') > u_4(\delta)$.

Hence, agent 4 has successfully manipulated, which contradicts strategyproofness. □

Remark 2. Theorem 1 strengthens a result of Bogomolnaia et al. [2005, Prop. 6] who proved the same statement with individual fair share (which requires that $u_i(f(\mathcal{A})) \geq C_i$) instead of positive share (i.e., $u_i(f(\mathcal{A})) > 0$), and assuming that $m \geq 17$ and $n \geq 5$. Their proof was substantially more complicated. It also strengthens the main result of Duddy [2015] who showed that no anonymous and neutral rule can be efficient, strategyproof, and decomposable when $m \geq 5$ and $n \geq 4$.

Remark 3. The bounds on m and n are tight for Theorem 1 to hold. Suppose that either $m < 4$ or $n < 5$. Then every *ex post* efficient distribution is also *ex ante* efficient [Duddy, 2015, Lemmas 1 and 2], which implies that *CUT* satisfies all conditions of Theorem 1.

5.2 Impossibility Without Symmetry Axioms

Theorem 1 answers one part of the open question posed by Bogomolnaia et al. [2005]: there is an impossibility involving positive share. The second part of their question was whether the impossibility holds even without requiring the symmetry axioms of anonymity and neutrality. While symmetry seems both desirable and mild, these axioms are actually rather restrictive in combination with the other axioms, as the reasoning in the proof of Theorem 1 makes clear. In practice, a distribution rule that occasionally violates anonymity and neutrality "by some ϵ" in order to satisfy other axioms could well be acceptable. Also, in some applications, there may be hierarchies or other asymmetries among agents or projects. It thus seems important to know whether the impossibility hinges on the symmetry assumption. Yet, Bogomolnaia et al. [2005] have "not been able to determine if one of the anonymity or neutrality property (or both) can be dropped", and Duddy [2015] noted that he "must concede that, like Bogomolnaia et al. [2005], we have been unable to demonstrate that all five properties are logically independent" in his related result (see Remark 2). We are now able to confirm that the impossibility holds without the symmetry axioms.

Theorem 2. No distribution rule satisfies efficiency, strategyproofness, and positive share when $m \geq 4$ and $n \geq 6$.

Our approach for obtaining impossibility theorems such as Theorems 1 and 2 is to use automated solvers to search for distribution rules that satisfy a list of axioms. In case the solver reports that our problem is infeasible, we have an impossibility that we can further analyze using minimal unsatisfiable set of constraints, which can often be translated to a human-readable proof of the result. This approach has been employed successfully to prove a number of impossibility theorems in social choice theory [see, e.g., Brandl et al., 2018, Geist and Endriss, 2011, Tang and Lin, 2009].

For our specific problem, using linear programming seems promising on first sight. We introduce a continuous variable $0 \leq z_{\mathcal{A},x} \leq 1$ for each profile $\mathcal{A}$ and each project $x \in A$ with the constraint $\sum_{x\in A} z_{\mathcal{A},x} = 1$ for each $\mathcal{A}$, so that these variables encode the output of a distribution rule. We can then easily add constraints to enforce strategyproofness and positive share (as well as anonymity and neutrality). However, it is not possible to enforce efficiency using linear programming, for example because the set of efficient distributions may not be convex [Aziz et al., 2015]. However, using a technique of Brandl et al. [2018, Sec. 4.2.2] that we describe in more detail below, efficiency can be enforced by introducing binary variables into the program, turning it into a mixed integer linear program. (Alternatively, using the same ideas we can encode the problem as an SMT formulation in the theory of linear arithmetic [see Brandl et al., 2018].) Unfortunately, the size of these formulations quickly becomes too large to be solved by current ILP and SMT solvers in reasonable time. For $m = 4$ and $n = 6$, the parameters used in Theorem 2, there are $15^6 \approx 11$ million different profiles (with unit contributions), and even if we only consider profiles up to reordering of agents (i.e., enforcing anonymity), there are more than 27 000 such profiles, still too many in practice.

Most work using this computer-aided approach has focussed on deterministic voting rules and has used SAT solvers [e.g., Brandt and Geist, 2016, Brandt et al., 2017, Kluiving et al., 2020]. The survey by Geist and Peters [2017] gives a tutorial of this method and explains the kind of encoding into conjunctive normal form that we use. SAT solvers can handle much larger problem instances. For example, Brandt et al. [2017, p. 23] report that they solved an instance based on 1.2 million profiles. Can we somehow discretize our problem so that we can use SAT solvers?

It turns out that we can, by only considering the *support* of the distribution returned by our distribution rule. Thus, our decision variables only need to keep track which projects are allocated a positive amount and which are allocated nothing. For 4 projects, there are only $2^4 - 1 = 15$ possible outcomes per profile (rather than the infinitely many distributions). Clearly, the positive share axiom only refers to the support, so it can easily be encoded in terms of these variables. Less obviously, Aziz et al. [2015] and Duddy [2015] have proved that whether a distribution is efficient or not depends only on its support, and one can compute the supports corresponding to efficient distributions via linear progamming in polynomial time [Aziz et al., 2015, Thm. 4]. The only remaining axiom is strategyproofness, which does depend on the precise distributions returned by the distribution rule. However, there are weakened versions of the strategyproofness axiom that can be phrased only in terms of supports. In particular, this is possible when we only consider clear-cut manipulations in which the manipulator enforces a distribution in which the *entire* endowment is distributed across her approved projects, so that by manipulating she obtains the maximum utility of C.

Definition 9 (Pessimistic strategyproofness)**.** A distribution rule is *pessimistically strategyproof*[7] if for all $i \in N$ and profiles $\mathcal{A}$ and $\mathcal{A}'$ with $A_j = A'_j$ for $j \neq i$, either $u_i(f(\mathcal{A})) = C$ or $u_i(f(\mathcal{A}')) < C$.

Clearly, strategyproofness implies pessimistic strategyproofness. It turns out that the impossibility still holds with the substantially weakened axiom that only forbids these clear-cut manipulations.

Even after discretizing, the formulas involved are very big, and further reduction techniques are needed. There are $15^6 \approx 11$ million different profiles with $n = 6$ and $m = 4$, and we need to use up to 15 variables for each profile (one for each allowed support), giving 170 million variables in total. It is much easier to obtain a result when we impose anonymity and neutrality. When we consider anonymous and neutral distribution rules, we only need to consider essentially different profiles (that are not equivalent up to renaming alternatives and agents), of which there are only 2197. In fact, as we saw in Theorem 1, with these extra axioms, the impossibility holds even for $n = 5$, for which there are only 736 essentially different profiles. Solving the resulting formula is almost instantaneous with a modern SAT solver. After extracting a minimal unsatisfiable set, we were astonished to find that it only referred to two different profiles, giving the short and elegant proof of Theorem 1. Notice that the proof only uses pessimistic strategyproofness.

So how can we obtain Theorem 2 in its full strength? We decided to remove anonymity and neutrality one at a time. First we dropped neutrality and verified that there exists no anonymous rule that is efficient, strategyproof, and satisfies positive share. This was feasible since there are 38 760 profiles that are not equivalent up to reordering agents, many fewer than the complete set of 11 million. The resulting formula with roughly 77 000 variables and 1 million clauses can be encoded in about 3 minutes using a python script, and can be solved in less than 1 second using a modern SAT solver such as lingeling [Biere et al., 2020]. A minimal unsatisfiable set (MUS) can be extracted using MUSer2 [Belov and Marques-Silva, 2012] in less than 3 seconds. We obtained an MUS that contained clauses referring to only 81 different profiles. We next constructed a formula using neither anonymity nor neutrality but which only referred to the 81 profiles we obtained in the last step plus all profiles obtained from them by permuting the $n = 6$ agents, for a total of $81 \cdot 6! \approx 58\,000$ profiles. We got lucky: the formula was unsatisfiable, and we found an MUS with about 500 different profiles. At this stage, we knew that Theorem 2 was true in its full strength.

[7] The term *pessimistic* is taken from the literature on manipulability of irresolute social choice functions [Duggan and Schwartz, 2000, Taylor, 2002]. If we view distributions as lotteries, then a pessimistic agent would evaluate a distribution by the utility of the worst possible outcome. A pessimistically strategyproof rule cannot be manipulated by a pessimistic agent. The dual notion of *optimistic strategyproofness* prevents agents from manipulating in profiles where their utility is 0. This kind of manipulation is impossible when positive share holds. Hence, positive share implies optimistic strategyproofness.

Finding an MUS that is human interpretable was an additional challenge. The first MUSes we found implicitly contained an intractable number of case analyses (while the final proof of Theorem 2, discussed below, only needs to distinguish two cases). We proceeded by first deriving a human-readable proof from the MUS for the case assuming anonymity, and then deriving a proof without anonymity from this intermediate step.

Proof of Theorem 2. The proof is long, and so the details are deferred to Appendix A, which consists of a description in Appendix A.1 of how to read the proof, and a listing of all proof steps in Appendices A.2 and A.3. Here, we give the high-level approach. We assume $m = 4$ and $n = 6$, and the proof can be adapted to larger values as before. Let f be an efficient and pessimistically strategyproof distribution rule satisfying positive share. The proof starts with the profile

$$\mathcal{A} = (\{b\}, \{c\}, \{a, b\}, \{a, c\}, \{b, d\}, \{c, d\}).$$

In $\mathcal{A}$, the only supports that are both efficient and satisfy positive share are $\{b, c\}$, $\{a, b, c\}$, and $\{b, c, d\}$. We proceed by case analysis on the support of the distribution $f(\mathcal{A})$. If the support is either $\{b, c\}$ or $\{a, b, c\}$, we follow the 192 steps displayed in Appendix A.2 to conclude that the support of $f(\mathcal{A})$ is actually $\{b, c, d\}$, a contradiction. The other case is that the support of $f(\mathcal{A})$ is $\{b, c, d\}$, in which case we follow the 196 steps in Appendix A.3 to conclude that the support of $f(\mathcal{A})$ is actually $\{a, b, c\}$, another contradiction. Hence such a rule f cannot exist. □

Remark 4. The axioms in Theorem 2 are independent. *UTIL* is efficient and strategyproof, *CUT* is strategyproof and satisfies positive share, and *NASH* is efficient and satisfies positive share.

Remark 5. The bounds on m and n are tight for Theorem 2 to hold. As discussed in Remark 3 if either $m < 4$ or $n < 5$ then *CUT* satisfies all conditions. For $m = 4$ and $n = 5$, one can modify the output of *CUT* on 96 profiles to get an anonymous, efficient, strategyproof, and decomposable rule.

Remark 6. Theorem 2 remains intact when weakening positive share so that it only applies to agents who approve a *single* project. This holds because an agent who receives utility 0 while approving more than one project can manipulate by narrowing her approval set to a single project.

Remark 7. Theorem 2 implies there is no efficient and strategyproof distribution rule which approximates egalitarian welfare (i.e., $\min_{i \in N} u_i(\delta)$), since the best attainable egalitarian welfare in any profile is always at least $\min_{i \in N} C_i > 0$, and we show that every efficient and strategyproof distribution rule will sometimes return a distribution with egalitarian welfare 0.

5.3 Impossibility for Subset Manipulations

Distribution rules satisfying notions such as positive share or decomposability try to be "fair" to each agent, and aim for an outcome that makes every agent reasonably happy. There is an obvious strategy to try to exploit this tendency: agents may pretend to be less happy than they are. In our setting, this would correspond to approving fewer projects.[8]

Definition 10 (Subset strategyproofness)**.** A distribution rule f is *subset strategyproof* if for any two profiles $\mathcal{A}$ and $\mathcal{A}'$ with $A_i' \subset A_i$ and $A_j = A_j'$ for all $j \neq i$, we have $u_i(f(\mathcal{A})) \geq u_i(f(\mathcal{A}'))$.

We can show, by a proof similar to the proof of Theorem 1, that every efficient distribution rule that satisfies positive share fails even subset strategyproofness. The proof appears in Appendix B. It uses anonymity and neutrality and, in contrast to Theorem 2, we do not know whether these can

[8] This notion of subset strategyproofness has also been studied in the context of proportional multiwinner elections [Peters, 2018]. The corresponding notion of *superset* strategyproofness has been studied by Aziz et al. [2019], who found that the egalitarian rule maximizing leximin welfare satisfies it, while *NASH* does not.

be dropped. The SAT solver indicates that neutrality cannot be dropped for $n \leq 7$ and $m = 5$ nor for $n \leq 6$ and $m = 6$. As before, the proof only uses the pessimistic version of subset strategyproofness.

Theorem 3. No anonymous and neutral distribution rule satisfies efficiency, subset strategyproofness, and positive share when $m \geq 5$ and $n \geq 5$.

6 CONCLUSION

We studied a model in which agents can vote on how to distribute an endowment over projects. The endowment may be given exogenously or obtained from contributions made by the agents. Which properties of distribution rules are desirable depends on the interpretation. For example, if the members of a community vote on how to allocate a fixed budget to public projects, the chosen distribution should fairly represent all agents. Whereas if philanthropically-minded donors team up to allocate their donations to charities more efficiently, the distribution rule should incentivize donors to contribute to the joint pool rather than to donate individually.

Bogomolnaia et al. [2005] considered the fixed-endowment case (with uniform contributions) and studied which distribution rules are efficient, strategyproof, and fair. Any two of these properties can be satisfied simultaneously, as the rules *UTIL*, *CUT*, and *NASH* exemplify (see Table 2). But our main result, Theorem 2, shows that it is impossible to satisfy all three properties (even for the weak notions of positive share and pessimistic strategyproofness), confirming a conjecture by Bogomolnaia et al. [2005]. The proof of this result, which reasons over several hundred profiles, is unlikely to have been found without the help of computers. When additionally assuming anonymity and neutrality, we have provided a simple proof with just two profiles.

We have introduced the sequential utilitarian rule *SUT*, which to our knowledge is the first rule known rule to satisfy efficiency, decomposability, and monotonicity. It does not satisfy weak participation, however. On the other hand, *CUT* and *NASH* are contribution incentive-compatible and satisfy monotonicity and efficiency, respectively. Interestingly, the deficiencies of each of these rules seem to be limited in practice: *SUT* rarely violates contribution incentive-compatibility, *NASH* monotonicity failures appear to be marginal, and simulations by Aziz et al. [2019] suggest that *CUT*'s efficiency failures are insignificant.[9] We leave as an open problem whether any distribution rule can satisfy all three of these axioms. This question is unlikely to be settled via computer, because *SUT* satisfies all three even for rather large profiles.

	UTIL	*CUT*	*NASH*	*SUT*	No Rule!
Efficiency	✓	–	✓	✓	↯
↳ Decomposable Efficiency	✓	✓	✓	✓	
Decomposability (Group Fair Share)	–	✓	✓	✓	
↳ Positive Share	–	✓	✓	✓	↯
Strategyproofness	✓	✓	–	–	↯
↳ Monotonicity	✓	✓	–	✓	
Contribution Incentive-Compatibility	–	✓	✓	–	
↳ Weak Participation	✓	✓	✓	–	

Table 2. Axiomatic properties of distribution rules.

[9] Also note that, even though the *CUT* distribution may be inefficient, it can never be dominated by the distributions returned by *NASH*, *SUT*, or any other decomposable rule. This is due to *CUT* satisfying decomposable efficiency.

REFERENCES

S. Airiau, H. Aziz, I. Caragiannis, J. Kruger, J. Lang, and D. Peters. Portioning using ordinal preferences: Fairness and efficiency. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence (IJCAI)*. IJCAI, 2019.

H. Aziz and A. Ganguly. Participatory funding coordination: Model, axioms and rules. Technical report, https://arxiv.org/pdf/2101.11722.pdf, 2021.

H. Aziz and N. Shah. Participatory budgeting: Models and approaches. In T. Rudas and P. Gábor, editors, *Pathways between Social Science and Computational Social Science: Theories, Methods and Interpretations*. Springer, 2020.

H. Aziz, F. Brandl, and F. Brandt. Universal Pareto dominance and welfare for plausible utility functions. *Journal of Mathematical Economics*, 60:123–133, 2015.

H. Aziz, B. E. Lee, and N. Talmon. Proportionally representative participatory budgeting: Axioms and algorithms. In *Proceedings of the 17th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 23–31, 2018.

H. Aziz, A. Bogomolnaia, and H. Moulin. Fair mixing: the case of dichotomous preferences. In *Proceedings of the 20th ACM Conference on Economics and Computation (ACM-EC)*, pages 753–781, 2019.

A. Belov and J. Marques-Silva. MUSer2: An efficient MUS extractor. *Journal on Satisfiability, Boolean Modeling and Computation*, 8:123–128, 2012.

G. Benade, S. Nath, A. D. Procaccia, and N. Shah. Preference elicitation for participatory budgeting. In *Proceedings of the 31st AAAI Conference on Artificial Intelligence (AAAI)*, pages 376–382, 2017.

A. Biere, K. Fazekas, M. Fleury, and M. Heisinger. CaDiCaL, Kissat, Paracooba, Plingeling and Treengeling entering the SAT Competition 2020. In T. Balyo, N. Froleyks, M. Heule, M. Iser, M. Järvisalo, and M. Suda, editors, *Proc. of SAT Competition 2020 – Solver and Benchmark Descriptions*, volume B-2020-1 of *Department of Computer Science Report Series B*, pages 51–53. University of Helsinki, 2020.

A. Bogomolnaia, H. Moulin, and R. Stong. Collective choice under dichotomous preferences. Mimeo, 2002.

A. Bogomolnaia, H. Moulin, and R. Stong. Collective choice under dichotomous preferences. *Journal of Economic Theory*, 122(2):165–184, 2005.

D. Bokal, B. Bresar, and J. Jerebic. A generalization of Hungarian method and Hall's theorem with applications in wireless sensor networks. *Discrete Applied Mathematics*, 160(4–5):460–470, 2012.

S. J. Brams and P. C. Fishburn. *Approval Voting*. Birkhäuser Boston, 1983.

F. Brandl, F. Brandt, and J. Hofbauer. Incentives for participation and abstention in probabilistic social choice. In *Proceedings of the 14th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 1411–1419, 2015.

F. Brandl, F. Brandt, and H. G. Seedig. Consistent probabilistic social choice. *Econometrica*, 84(5):1839–1880, 2016.

F. Brandl, F. Brandt, M. Eberl, and C. Geist. Proving the incompatibility of efficiency and strategyproofness via SMT solving. *Journal of the ACM*, 65(2):1–28, 2018.

F. Brandl, F. Brandt, D. Peters, C. Stricker, and W. Suksompong. Funding public projects: A case for the Nash product rule. 2020. Working paper.

F. Brandt. Rolling the dice: Recent results in probabilistic social choice. In U. Endriss, editor, *Trends in Computational Social Choice*, chapter 1, pages 3–26. AI Access, 2017.

F. Brandt and C. Geist. Finding strategyproof social choice functions via SAT solving. *Journal of Artificial Intelligence Research*, 55:565–602, 2016.

F. Brandt, C. Geist, and D. Peters. Optimal bounds for the no-show paradox via SAT solving. *Mathematical Social Sciences*, 90:18–27, 2017. Special Issue in Honor of Hervé Moulin.

Y. Cabannes. Participatory budgeting: a significant contribution to participatory democracy. *Environment and Urbanization*, 16(1):27–46, 2004.

G. Carroll. An efficiency theorem for incompletely known preferences. *Journal of Economic Theory*, 145(6):2463–2470, 2010.

C. Duddy. Fair sharing under dichotomous preferences. *Mathematical Social Sciences*, 73:1–5, 2015.

J. Duggan and T. Schwartz. Strategic manipulability without resoluteness or shared beliefs: Gibbard-Satterthwaite generalized. *Social Choice and Welfare*, 17(1):85–93, 2000.

B. Fain, A. Goel, and K. Munagala. The core of the participatory budgeting problem. In *Proceedings of the 12th International Conference on Web and Internet Economics (WINE)*, Lecture Notes in Computer Science (LNCS), pages 384–399. Springer-Verlag, 2016.

B. Fain, K. Munagala, and N. Shah. Fair allocation of indivisible public goods. In *Proceedings of the 19th ACM Conference on Economics and Computation (ACM-EC)*, pages 575–592, 2018.

D. K. Foley. Lindahl's solution and the core of an economy with public goods. *Econometrica*, 38(1):66–72, 1970.

R. Freeman, D. M. Pennock, D. Peters, and J. W. Vaughan. Truthful aggregation of budget proposals. In *Proceedings of the 2019 ACM Conference on Economics and Computation (ACM-EC)*, pages 751–752, 2019.

C. Geist and U. Endriss. Automated search for impossibility theorems in social choice theory: Ranking sets of objects. *Journal of Artificial Intelligence Research*, 40:143–174, 2011.

C. Geist and D. Peters. Computer-aided methods for social choice theory. In U. Endriss, editor, *Trends in Computational Social Choice*, chapter 13, pages 249–267. AI Access, 2017.

A. Gibbard. Manipulation of schemes that mix voting with chance. *Econometrica*, 45(3):665–681, 1977.

A. Goel, A. K. Krishnaswamy, S. Sakshuwong, and T. Aitamurto. Knapsack voting for participatory budgeting. *ACM Transactions on Economics and Computation*, 7(2):8:1–8:27, 2019.

A. Guerdjikova and K. Nehring. Weighing experts, weighing sources: The diversity value. 2014. Mimeo.

A. Hylland. Strategyproofness of voting procedures with lotteries as outcomes and infinite sets of strategies. Mimeo, 1980.

B. Kluiving, A. de Vries, P. Vrijbergen, A. Boixel, and U. Endriss. Analysing irresolute multiwinner voting rules with approval ballots via SAT solving. In *Proceedings of the 24th European Conference on Artificial Intelligence (ECAI)*, 2020.

M. Michorzewski, D. Peters, and P. Skowron. Price of fairness in budget division and probabilistic social choice. In *Proceedings of the 34th AAAI Conference on Artificial Intelligence (AAAI)*, 2020. Forthcoming.

H. Moulin. *Axioms of Cooperative Decision Making*. Cambridge University Press, 1988a.

H. Moulin. Condorcet's principle implies the no show paradox. *Journal of Economic Theory*, 45(1):53–64, 1988b.

D. Peters. Proportionality and strategyproofness in multiwinner elections. In *Proceedings of the 17th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, volume 1549–1557, 2018.

N. Talmon and P. Faliszewski. A framework for approval-based budgeting methods. In *Proceedings of the 33rd AAAI Conference on Artificial Intelligence (AAAI)*, pages 2181–2188, 2019.

P. Tang and F. Lin. Computer-aided proofs of Arrow's and other impossibility theorems. *Artificial Intelligence*, 173(11): 1041–1053, 2009.

Z. Tang, C. Wang, and M. Zhang. Price of fairness in budget division for egalitarian social welfare. In *Proceedings of the 14th International Conference on Combinatorial Optimization and Applications (COCOA)*, pages 594–607. Springer, 2020.

A. D. Taylor. The manipulability of voting systems. *The American Mathematical Monthly*, 109(4):321–337, 2002.

A FULL PROOF OF THEOREM 2

Theorem. No distribution rule satisfies efficiency, (pessimistic) strategyproofness, and positive share when $m \geq 4$ and $n \geq 6$.

We assume $m = 4$ and $n = 6$, and the proof can be adapted to larger values as in Theorem 1 by adding agents approving all projects or by adding projects which no-one approves. Let f be an efficient and pessimistically strategyproof distribution rule satisfying positive share. We consider the profile with uniform contributions

$$\mathcal{A}_1 = (\{b\}, \{c\}, \{a, b\}, \{a, c\}, \{b, d\}, \{c, d\}),$$

which we will also call "Profile 1". For this profile, all supports that contain both a and d are dominated by supports that contain b and c. Hence, the only supports that are both efficient and satisfy positive share are $\{b, c\}$, $\{a, b, c\}$, and $\{b, c, d\}$. We proceed by case analysis on the support of the distribution $f(\mathcal{A}_1)$. If the support is either $\{b, c\}$ or $\{a, b, c\}$, we follow the 192 steps displayed in Appendix A.2 to conclude that the support of $f(\mathcal{A}_1)$ is actually $\{b, c, d\}$, a contradiction. The other case is that the support of $f(\mathcal{A}_1)$ is $\{b, c, d\}$, in which case we follow the 196 steps in Appendix A.3 to conclude that the support of $f(\mathcal{A}_1)$ is actually $\{a, b, c\}$, another contradiction. Hence such a rule f cannot exist.

A.1 How to read the proof

To save space, for the rest of the proof, we will omit set braces and commas, so for example we write *abc* for $\{a, b, c\}$.

Appendices A.2 and A.3 each display a list of profiles, all with uniform contributions. The last column lists all "possible supports" for that profile, i.e., all non-empty subsets $A' \subseteq A$ such that (i) every distribution with support A' is efficient and (ii) each agent approves at least one project in A'. By assumption, f must select a distribution whose support is among the possible supports.

We computed the list of possible supports by going through all supports A' satisfying positive share and then checking whether another support A'' dominates it. We say that A'' dominates A' if the uniform distribution $\delta'' = \sum_{x \in A''} 1/|A''|$ over A'' dominates the uniform distribution $\delta' = \sum_{x \in A'} 1/|A'|$ over A'. In this case, we write $A' \leftarrowtail A''$. For example, in profile $\mathcal{A}_1$, the distribution $\frac{1}{2}b + \frac{1}{2}c$ dominates $\frac{1}{2}a + \frac{1}{2}d$, and hence $ad \leftarrowtail bc$. For our problem, it was enough to only consider pairs A', A'' of equal cardinality. If we eliminate support A' because it is dominated, we then also eliminate all supersets of A' because those are also dominated. The following lists of profiles mention dominated supports for ease of verification.

In the first row of the list, which refers to Profile 1, we have underlined the support(s) that we have assumed are used by $f(\mathcal{A}_1)$. From here, we read the list from top to bottom: Profile $k + 1$ is obtained from Profile k by replacing the approval set of exactly one voter. The proof now establishes step by step that if f selects one of the supports underlined for Profile k, then it must select one of the supports underlined for Profile $k + 1$. This is done by invoking pessimistic strategyproofness at each step. There are two ways to invoke strategyproofness: Suppose Profiles k and $k + 1$ differ only in the report of agent $i \in N$. Then we can either use strategyproofness by considering a manipulation by i from Profile k to $k + 1$, or a manipulation by i from Profile $k + 1$ to k. The preferences of the voter who changed her preferences are highlighted in gray.

Let us work through an example. In Appendix A.2, write $\mathcal{A}_k$ for Profile k. We assume that the support of $f(\mathcal{A}_1)$ is either *bc* or *abc*. Consider Profile 2. Possible supports in this profile are *bc* and *bcd*. Suppose the support of $f(\mathcal{A}_2)$ is *bcd*. Then agent 3 is not completely satisfied because agent 3 does not approve d. Thus, agent 3 can manipulate to Profile 1, where the selected support (*bc* or *abc*) only contains projects approved by agent 3 in Profile 2. Hence *bcd* cannot be the support of

$f(\mathcal{A}_2)$. Thus the support is *bc*, which we can therefore underline. The argument why $f(\mathcal{A}_3)$ must have support *bc* as well is identical.

For another example, consider Profile 10, for which we have deduced from prior steps that $f(\mathcal{A}_{10})$ has support *cd* or *acd*. Thus voter 4 (with approval set *abc*) is not entirely satisfied, since voter 4 does not approve *d*. In Profile 11, the possible supports are *ac*, *acd*, and *abc*. If $f(\mathcal{A}_{11})$ had support *ac* or *abc*, then voter 4 could manipulate from Profile 10 to Profile 11 and obtain a distribution that only uses approved alternatives. Hence the support must be *acd*, which we can therefore underline.

We repeat this type of argument 192 times to find that $f(\mathcal{A}_1)$ has support *bcd*, a contradiction.

A.2 Assuming $f(\mathcal{A}_1)$ has support *bc* or *abc* leads to contradiction.

	A_1	A_2	A_3	A_4	A_5	A_6	possible supports	dominated supports
Profile 1	*b*	*c*	*ab*	*ac*	*bd*	*cd*	$\underline{bc}, \underline{abc}, bcd$	$ad \leftarrowtail bc$
Profile 2	*b*	*c*	*abc*	*ac*	*bd*	*cd*	$\underline{bc}, bcd$	$a \leftarrowtail c$, $ab \leftarrowtail bc$, $ad \leftarrowtail bc$
Profile 3	*b*	*c*	*bc*	*ac*	*bd*	*cd*	$\underline{bc}, bcd$	$a \leftarrowtail c$, $ab \leftarrowtail bc$, $ad \leftarrowtail bc$
Profile 4	*bc*	*c*	*bc*	*ac*	*bd*	*cd*	$cd, \underline{bc}, bcd$	$a \leftarrowtail c$, $ab \leftarrowtail bc$, $ad \leftarrowtail bc$
Profile 5	*bc*	*c*	*bc*	*ac*	*bd*	*acd*	$cd, \underline{bc}, \underline{bcd}$	$a \leftarrowtail c$, $ab \leftarrowtail bc$, $ad \leftarrowtail cd$
Profile 6	*bc*	*c*	*bc*	*ac*	*bd*	*ad*	$cd, acd, \underline{bcd}$	$ab \leftarrowtail cd$
Profile 7	*bc*	*c*	*bc*	*ac*	*bcd*	*ad*	$ac, \underline{cd}, acd$	$b \leftarrowtail c$, $ab \leftarrowtail ac$, $bd \leftarrowtail cd$
Profile 8	*bc*	*c*	*bc*	*ac*	*cd*	*ad*	$ac, \underline{cd}, acd$	$b \leftarrowtail c$, $ab \leftarrowtail ac$, $bd \leftarrowtail ac$
Profile 9	*bc*	*c*	*cd*	*ac*	*cd*	*ad*	$ac, \underline{cd}, acd$	$b \leftarrowtail c$, $ab \leftarrowtail ac$, $bd \leftarrowtail ac$
Profile 10	*bc*	*c*	*cd*	*abc*	*cd*	*ad*	$ac, \underline{cd}, \underline{acd}$	$b \leftarrowtail c$, $ab \leftarrowtail ac$, $bd \leftarrowtail ac$
Profile 11	*bc*	*c*	*cd*	*ab*	*cd*	*ad*	$ac, \underline{acd}, abc$	$bd \leftarrowtail ac$
Profile 12	*bc*	*c*	*cd*	*ab*	*cd*	*acd*	$\underline{ac}, bc, abc$	$d \leftarrowtail c$, $ad \leftarrowtail ac$, $bd \leftarrowtail ac$
Profile 13	*bc*	*c*	*cd*	*ab*	*cd*	*ac*	$\underline{ac}, bc, abc$	$d \leftarrowtail c$, $ad \leftarrowtail ac$, $bd \leftarrowtail ac$
Profile 14	*bcd*	*c*	*cd*	*ab*	*cd*	*ac*	$\underline{ac}, bc, \underline{abc}$	$d \leftarrowtail c$, $ad \leftarrowtail ac$, $bd \leftarrowtail bc$
Profile 15	*bd*	*c*	*cd*	*ab*	*cd*	*ac*	$bc, \underline{abc}, bcd$	$ad \leftarrowtail bc$
Profile 16	*bd*	*c*	*cd*	*abc*	*cd*	*ac*	$cd, \underline{bc}, bcd$	$a \leftarrowtail c$, $ab \leftarrowtail bc$, $ad \leftarrowtail bc$
Profile 17	*bd*	*c*	*cd*	*bc*	*cd*	*ac*	$cd, \underline{bc}, bcd$	$a \leftarrowtail c$, $ab \leftarrowtail bc$, $ad \leftarrowtail bc$
Profile 18	*bd*	*c*	*acd*	*bc*	*cd*	*ac*	$cd, \underline{bc}, \underline{bcd}$	$a \leftarrowtail c$, $ab \leftarrowtail bc$, $ad \leftarrowtail cd$
Profile 19	*bd*	*c*	*acd*	*bc*	*d*	*ac*	$cd, \underline{bc}, \underline{bcd}$	$a \leftarrowtail c$, $ab \leftarrowtail bc$, $ad \leftarrowtail cd$
Profile 20	*bd*	*c*	*ad*	*bc*	*d*	*ac*	$cd, acd, \underline{bcd}$	$ab \leftarrowtail cd$
Profile 21	*bd*	*c*	*ad*	*bcd*	*d*	*ac*	$\underline{cd}, acd$	$b \leftarrowtail d$, $ab \leftarrowtail ad$, $bc \leftarrowtail cd$
Profile 22	*bd*	*cd*	*ad*	*bcd*	*d*	*ac*	$ad, \underline{cd}, acd$	$b \leftarrowtail d$, $ab \leftarrowtail ad$, $bc \leftarrowtail cd$
Profile 23	*bd*	*cd*	*ad*	*cd*	*d*	*ac*	$ad, \underline{cd}, acd$	$b \leftarrowtail d$, $ab \leftarrowtail ad$, $bc \leftarrowtail ad$
Profile 24	*bd*	*cd*	*abd*	*cd*	*d*	*ac*	$ad, \underline{cd}, \underline{acd}$	$b \leftarrowtail d$, $ab \leftarrowtail ad$, $bc \leftarrowtail ad$
Profile 25	*bd*	*cd*	*ab*	*cd*	*d*	*ac*	$\underline{ad}, abd, \underline{acd}$	$bc \leftarrowtail ad$
Profile 26	*bd*	*cd*	*ab*	*cd*	*d*	*acd*	$\underline{ad}, bd, abd$	$c \leftarrowtail d$, $ac \leftarrowtail ad$, $bc \leftarrowtail ad$
Profile 27	*bd*	*cd*	*ab*	*cd*	*d*	*ad*	$\underline{ad}, bd, abd$	$c \leftarrowtail d$, $ac \leftarrowtail ad$, $bc \leftarrowtail ad$
Profile 28	*bcd*	*cd*	*ab*	*cd*	*d*	*ad*	$\underline{ad}, bd, \underline{abd}$	$c \leftarrowtail d$, $ac \leftarrowtail ad$, $bc \leftarrowtail bd$
Profile 29	*bc*	*cd*	*ab*	*cd*	*d*	*ad*	$\underline{bd}, \underline{abd}, bcd$	$ac \leftarrowtail bd$

Profile 30	bc	cd	ab	cd	d	abd	$\underline{bd}, bcd$	$a \leftarrowtail b,\ ac \leftarrowtail bc,\ ad \leftarrowtail bd$
Profile 31	bc	cd	ab	cd	d	bd	$\underline{bd}, bcd$	$a \leftarrowtail b,\ ac \leftarrowtail bc,\ ad \leftarrowtail bd$
Profile 32	bc	cd	bd	cd	d	bd	$\underline{bd}, cd, bcd$	$a \leftarrowtail b,\ ab \leftarrowtail bc,\ ac \leftarrowtail bc,\ ad \leftarrowtail bd$
Profile 33	bc	cd	bd	ac	d	bd	$cd, \underline{bcd}$	$a \leftarrowtail c,\ ab \leftarrowtail bc,\ ad \leftarrowtail cd$
Profile 34	bc	acd	bd	ac	d	bd	$cd, \underline{bcd}$	$a \leftarrowtail c,\ ab \leftarrowtail bc,\ ad \leftarrowtail cd$
Profile 35	bc	ad	bd	ac	d	bd	$cd, acd, \underline{bcd}$	$ab \leftarrowtail cd$
Profile 36	bcd	ad	bd	ac	d	bd	$ad, \underline{cd}, acd$	$b \leftarrowtail d,\ ab \leftarrowtail ad,\ bc \leftarrowtail cd$
Profile 37	cd	ad	bd	ac	d	bd	$ad, \underline{cd}, acd$	$b \leftarrowtail d,\ ab \leftarrowtail ad,\ bc \leftarrowtail ad$
Profile 38	cd	abd	bd	ac	d	bd	$ad, \underline{cd}, \underline{acd}$	$b \leftarrowtail d,\ ab \leftarrowtail ad,\ bc \leftarrowtail ad$
Profile 39	cd	ab	bd	ac	d	bd	$ad, \underline{acd}, abd$	$bc \leftarrowtail ad$
Profile 40	cd	ab	bd	acd	d	bd	$\underline{ad}, bd, abd$	$c \leftarrowtail d,\ ac \leftarrowtail ad,\ bc \leftarrowtail ad$
Profile 41	cd	ab	bd	ad	d	bd	$\underline{ad}, bd, abd$	$c \leftarrowtail d,\ ac \leftarrowtail ad,\ bc \leftarrowtail ad$
Profile 42	cd	ab	bcd	ad	d	bd	$\underline{ad}, bd, \underline{abd}$	$c \leftarrowtail d,\ ac \leftarrowtail ad,\ bc \leftarrowtail bd$
Profile 43	cd	ab	bcd	ad	d	b	$bd, \underline{abd}$	$c \leftarrowtail d,\ ac \leftarrowtail ad,\ bc \leftarrowtail bd$
Profile 44	cd	ab	bc	ad	d	b	$bd, \underline{abd}, bcd$	$ac \leftarrowtail bd$
Profile 45	cd	ab	bc	abd	d	b	$\underline{bd}, bcd$	$a \leftarrowtail b,\ ac \leftarrowtail bc,\ ad \leftarrowtail bd$
Profile 46	cd	ab	bc	bd	d	b	$\underline{bd}, bcd$	$a \leftarrowtail b,\ ac \leftarrowtail bc,\ ad \leftarrowtail bd$
Profile 47	cd	ab	bc	bd	bd	b	$bc, \underline{bd}, bcd$	$a \leftarrowtail b,\ ac \leftarrowtail bc,\ ad \leftarrowtail bc$
Profile 48	cd	ab	abc	bd	bd	b	$bc, \underline{bd}, \underline{bcd}$	$a \leftarrowtail b,\ ac \leftarrowtail bc,\ ad \leftarrowtail bc$
Profile 49	cd	ab	ac	bd	bd	b	$bc, abc, \underline{bcd}$	$ad \leftarrowtail bc$
Profile 50	bcd	ab	ac	bd	bd	b	$ab, \underline{bc}, abc$	$d \leftarrowtail b,\ ad \leftarrowtail ab,\ cd \leftarrowtail bc$
Profile 51	bc	ab	ac	bd	bd	b	$ab, \underline{bc}, abc$	$d \leftarrowtail b,\ ad \leftarrowtail ab,\ cd \leftarrowtail ab$
Profile 52	bc	ab	ac	bc	bd	b	$ab, \underline{bc}, abc$	$d \leftarrowtail b,\ ad \leftarrowtail ab,\ cd \leftarrowtail ab$
Profile 53	bc	abd	ac	bc	bd	b	$ab, \underline{bc}, \underline{abc}$	$d \leftarrowtail b,\ ad \leftarrowtail ab,\ cd \leftarrowtail ab$
Profile 54	bc	ad	ac	bc	bd	b	$ab, \underline{abc}, abd$	$cd \leftarrowtail ab$
Profile 55	bc	ad	abc	bc	bd	b	$\underline{ab}, bd, abd$	$c \leftarrowtail b,\ ac \leftarrowtail ab,\ cd \leftarrowtail ab$
Profile 56	bc	ad	ab	bc	bd	b	$\underline{ab}, bd, abd$	$c \leftarrowtail b,\ ac \leftarrowtail ab,\ cd \leftarrowtail ab$
Profile 57	bc	ad	ab	bc	bcd	b	$\underline{ab}, bd, \underline{abd}$	$c \leftarrowtail b,\ ac \leftarrowtail ab,\ cd \leftarrowtail bd$
Profile 58	bc	ad	ab	bc	cd	b	$bd, \underline{abd}, bcd$	$ac \leftarrowtail bd$
Profile 59	bc	abd	ab	bc	cd	b	$bc, \underline{bd}, bcd$	$a \leftarrowtail b,\ ac \leftarrowtail bc,\ ad \leftarrowtail bd$
Profile 60	bc	bd	ab	bc	cd	b	$bc, \underline{bd}, bcd$	$a \leftarrowtail b,\ ac \leftarrowtail bc,\ ad \leftarrowtail bc$
Profile 61	bc	bd	ab	abc	cd	b	$bc, \underline{bd}, \underline{bcd}$	$a \leftarrowtail b,\ ac \leftarrowtail bc,\ ad \leftarrowtail bc$
Profile 62	c	bd	ab	abc	cd	b	$bc, \underline{bd}, \underline{bcd}$	$a \leftarrowtail b,\ ac \leftarrowtail bc,\ ad \leftarrowtail bc$
Profile 63	c	bd	ab	ac	cd	b	$bc, abc, \underline{bcd}$	$ad \leftarrowtail bc$
Profile 64	c	bcd	ab	ac	cd	b	$\underline{bc}, abc$	$d \leftarrowtail c,\ ad \leftarrowtail ac,\ bd \leftarrowtail bc$
Profile 65	c	bcd	ab	ac	cd	bc	$ac, \underline{bc}, abc$	$d \leftarrowtail c,\ ad \leftarrowtail ac,\ bd \leftarrowtail bc$
Profile 66	c	bc	ab	ac	cd	bc	$ac, \underline{bc}, abc$	$d \leftarrowtail c,\ ad \leftarrowtail ac,\ bd \leftarrowtail ac$
Profile 67	c	bc	ab	acd	cd	bc	$ac, \underline{bc}, \underline{abc}$	$d \leftarrowtail c,\ ad \leftarrowtail ac,\ bd \leftarrowtail ac$
Profile 68	c	bc	ab	ad	cd	bc	$\underline{ac}, acd, \underline{abc}$	$bd \leftarrowtail ac$
Profile 69	c	bc	abc	ad	cd	bc	$\underline{ac}, cd, acd$	$b \leftarrowtail c,\ ab \leftarrowtail ac,\ bd \leftarrowtail ac$

Profile 70	*c*	*bc*	*ac*	*ad*	*cd*	*bc*	*ac*, *cd*, *acd*	$b \leftarrowtail c,\ ab \leftarrowtail ac,\ bd \leftarrowtail ac$
Profile 71	*c*	*bc*	*ac*	*ad*	*bcd*	*bc*	*ac*, *cd*, *acd*	$b \leftarrowtail c,\ ab \leftarrowtail ac,\ bd \leftarrowtail cd$
Profile 72	*c*	*bc*	*ac*	*ad*	*bd*	*bc*	*cd*, *acd*, *bcd*	$ab \leftarrowtail cd$
Profile 73	*c*	*bc*	*acd*	*ad*	*bd*	*bc*	*cd*, *bcd*	$a \leftarrowtail d,\ ab \leftarrowtail bd,\ ac \leftarrowtail cd$
Profile 74	*c*	*bc*	*cd*	*ad*	*bd*	*bc*	*cd*, *bcd*	$a \leftarrowtail d,\ ab \leftarrowtail bd,\ ac \leftarrowtail cd$
Profile 75	*c*	*bc*	*cd*	*cd*	*bd*	*bc*	*cd*, *bc*, *bcd*	$a \leftarrowtail b,\ ab \leftarrowtail bc,\ ac \leftarrowtail bc,\ ad \leftarrowtail bc$
Profile 76	*c*	*ab*	*cd*	*cd*	*bd*	*bc*	*bc*, *bcd*	$a \leftarrowtail b,\ ac \leftarrowtail bc,\ ad \leftarrowtail bc$
Profile 77	*c*	*ab*	*cd*	*cd*	*bd*	*abc*	*bc*, *bcd*	$a \leftarrowtail b,\ ac \leftarrowtail bc,\ ad \leftarrowtail bc$
Profile 78	*c*	*ab*	*cd*	*cd*	*bd*	*ac*	*bc*, *abc*, *bcd*	$ad \leftarrowtail bc$
Profile 79	*c*	*ab*	*cd*	*cd*	*bcd*	*ac*	*ac*, *bc*, *abc*	$d \leftarrowtail c,\ ad \leftarrowtail ac,\ bd \leftarrowtail bc$
Profile 80	*c*	*ab*	*cd*	*cd*	*bc*	*ac*	*ac*, *bc*, *abc*	$d \leftarrowtail c,\ ad \leftarrowtail ac,\ bd \leftarrowtail ac$
Profile 81	*c*	*ab*	*cd*	*cd*	*bc*	*acd*	*ac*, *bc*, *abc*	$d \leftarrowtail c,\ ad \leftarrowtail ac,\ bd \leftarrowtail ac$
Profile 82	*c*	*ab*	*cd*	*cd*	*bc*	*ad*	*ac*, *abc*, *acd*	$bd \leftarrowtail ac$
Profile 83	*c*	*abc*	*cd*	*cd*	*bc*	*ad*	*ac*, *cd*, *acd*	$b \leftarrowtail c,\ ab \leftarrowtail ac,\ bd \leftarrowtail ac$
Profile 84	*c*	*ac*	*cd*	*cd*	*bc*	*ad*	*ac*, *cd*, *acd*	$b \leftarrowtail c,\ ab \leftarrowtail ac,\ bd \leftarrowtail ac$
Profile 85	*c*	*ac*	*bcd*	*cd*	*bc*	*ad*	*ac*, *cd*, *acd*	$b \leftarrowtail c,\ ab \leftarrowtail ac,\ bd \leftarrowtail cd$
Profile 86	*c*	*ac*	*bcd*	*d*	*bc*	*ad*	*cd*, *acd*	$b \leftarrowtail c,\ ab \leftarrowtail ac,\ bd \leftarrowtail cd$
Profile 87	*c*	*ac*	*bd*	*d*	*bc*	*ad*	*cd*, *acd*, *bcd*	$ab \leftarrowtail cd$
Profile 88	*c*	*acd*	*bd*	*d*	*bc*	*ad*	*cd*, *bcd*	$a \leftarrowtail d,\ ab \leftarrowtail bd,\ ac \leftarrowtail cd$
Profile 89	*c*	*cd*	*bd*	*d*	*bc*	*ad*	*cd*, *bcd*	$a \leftarrowtail d,\ ab \leftarrowtail bd,\ ac \leftarrowtail cd$
Profile 90	*cd*	*cd*	*bd*	*d*	*bc*	*ad*	*bd*, *cd*, *bcd*	$a \leftarrowtail d,\ ab \leftarrowtail bd,\ ac \leftarrowtail bd$
Profile 91	*cd*	*cd*	*abd*	*d*	*bc*	*ad*	*bd*, *cd*, *bcd*	$a \leftarrowtail d,\ ab \leftarrowtail bd,\ ac \leftarrowtail bd$
Profile 92	*cd*	*cd*	*ab*	*d*	*bc*	*ad*	*bd*, *abd*, *bcd*	$ac \leftarrowtail bd$
Profile 93	*cd*	*cd*	*ab*	*d*	*bcd*	*ad*	*ad*, *bd*, *abd*	$c \leftarrowtail d,\ ac \leftarrowtail ad,\ bc \leftarrowtail bd$
Profile 94	*cd*	*cd*	*ab*	*d*	*bd*	*ad*	*ad*, *bd*, *abd*	$c \leftarrowtail d,\ ac \leftarrowtail ad,\ bc \leftarrowtail ad$
Profile 95	*cd*	*bd*	*ab*	*d*	*bd*	*ad*	*ad*, *bd*, *abd*	$c \leftarrowtail d,\ ac \leftarrowtail ad,\ bc \leftarrowtail ad$
Profile 96	*cd*	*bd*	*ab*	*d*	*bd*	*acd*	*ad*, *bd*, *abd*	$c \leftarrowtail d,\ ac \leftarrowtail ad,\ bc \leftarrowtail ad$
Profile 97	*cd*	*bd*	*ab*	*d*	*bd*	*ac*	*ad*, *abd*, *acd*	$bc \leftarrowtail ad$
Profile 98	*cd*	*bd*	*abd*	*d*	*bd*	*ac*	*ad*, *cd*, *acd*	$b \leftarrowtail d,\ ab \leftarrowtail ad,\ bc \leftarrowtail ad$
Profile 99	*cd*	*bd*	*ad*	*d*	*bd*	*ac*	*ad*, *cd*, *acd*	$b \leftarrowtail d,\ ab \leftarrowtail ad,\ bc \leftarrowtail ad$
Profile 100	*bcd*	*bd*	*ad*	*d*	*bd*	*ac*	*ad*, *cd*, *acd*	$b \leftarrowtail d,\ ab \leftarrowtail ad,\ bc \leftarrowtail cd$
Profile 101	*bc*	*bd*	*ad*	*d*	*bd*	*ac*	*cd*, *acd*, *bcd*	$ab \leftarrowtail cd$
Profile 102	*bc*	*bd*	*ad*	*d*	*bd*	*acd*	*bd*, *cd*, *bcd*	$a \leftarrowtail d,\ ab \leftarrowtail bd,\ ac \leftarrowtail cd$
Profile 103	*bc*	*bd*	*ad*	*d*	*bd*	*cd*	*bd*, *cd*, *bcd*	$a \leftarrowtail d,\ ab \leftarrowtail bd,\ ac \leftarrowtail bd$
Profile 104	*bc*	*bd*	*ad*	*d*	*abd*	*cd*	*bd*, *cd*, *bcd*	$a \leftarrowtail d,\ ab \leftarrowtail bd,\ ac \leftarrowtail bd$
Profile 105	*bc*	*b*	*ad*	*d*	*abd*	*cd*	*bd*, *cd*, *bcd*	$a \leftarrowtail d,\ ab \leftarrowtail bd,\ ac \leftarrowtail bd$
Profile 106	*bc*	*b*	*ad*	*d*	*ab*	*cd*	*bd*, *abd*, *bcd*	$ac \leftarrowtail bd$
Profile 107	*bc*	*b*	*ad*	*d*	*ab*	*bcd*	*bd*, *abd*	$c \leftarrowtail b,\ ac \leftarrowtail ab,\ cd \leftarrowtail bd$
Profile 108	*bc*	*b*	*ad*	*bd*	*ab*	*bcd*	*ab*, *bd*, *abd*	$c \leftarrowtail b,\ ac \leftarrowtail ab,\ cd \leftarrowtail bd$
Profile 109	*bc*	*b*	*ad*	*bd*	*ab*	*bd*	*ab*, *bd*, *abd*	$c \leftarrowtail b,\ ac \leftarrowtail ab,\ cd \leftarrowtail ab$

Profile 110	bc	b	ad	bd	abc	bd	ab, bd, abd	c ↢ b, ac ↢ ab, cd ↢ ab
Profile 111	bc	b	ad	bd	ac	bd	ab, abc, abd	cd ↢ ab
Profile 112	bc	b	abd	bd	ac	bd	ab, bc, abc	d ↢ b, ad ↢ ab, cd ↢ ab
Profile 113	bc	b	ab	bd	ac	bd	ab, bc, abc	d ↢ b, ad ↢ ab, cd ↢ ab
Profile 114	bcd	b	ab	bd	ac	bd	ab, bc, abc	d ↢ b, ad ↢ ab, cd ↢ bc
Profile 115	cd	b	ab	bd	ac	bd	bc, abc, bcd	ad ↢ bc
Profile 116	cd	b	abc	bd	ac	bd	bc, bcd	a ↢ c, ab ↢ bc, ad ↢ bc
Profile 117	cd	b	bc	bd	ac	bd	bc, bcd	a ↢ c, ab ↢ bc, ad ↢ bc
Profile 118	cd	b	bc	bd	bc	bd	bc, bd, bcd	a ↢ b, ab ↢ bc, ac ↢ bc, ad ↢ bc
Profile 119	cd	b	bc	bd	bc	ad	bd, bcd	a ↢ d, ab ↢ bd, ac ↢ bd
Profile 120	cd	b	bc	abd	bc	ad	bd, bcd	a ↢ d, ab ↢ bd, ac ↢ bd
Profile 121	cd	b	bc	ab	bc	ad	bd, abd, bcd	ac ↢ bd
Profile 122	bcd	b	bc	ab	bc	ad	ab, bd, abd	c ↢ b, ac ↢ ab, cd ↢ bd
Profile 123	bd	b	bc	ab	bc	ad	ab, bd, abd	c ↢ b, ac ↢ ab, cd ↢ ab
Profile 124	bd	b	bc	abc	bc	ad	ab, bd, abd	c ↢ b, ac ↢ ab, cd ↢ ab
Profile 125	bd	b	bc	ac	bc	ad	ab, abd, abc	cd ↢ ab
Profile 126	bd	b	bc	ac	bc	abd	ab, bc, abc	d ↢ b, ad ↢ ab, cd ↢ ab
Profile 127	bd	b	bc	ac	bc	ab	ab, bc, abc	d ↢ b, ad ↢ ab, cd ↢ ab
Profile 128	bd	b	bcd	ac	bc	ab	ab, bc, abc	d ↢ b, ad ↢ ab, cd ↢ bc
Profile 129	bd	b	bcd	ac	c	ab	bc, abc	d ↢ b, ad ↢ ab, cd ↢ bc
Profile 130	bd	b	cd	ac	c	ab	bc, abc, bcd	ad ↢ bc
Profile 131	bd	b	cd	ac	c	abc	bc, bcd	a ↢ c, ab ↢ bc, ad ↢ bc
Profile 132	bd	b	cd	ac	c	bc	bc, bcd	a ↢ c, ab ↢ bc, ad ↢ bc
Profile 133	bd	bc	cd	ac	c	bc	cd, bc, bcd	a ↢ c, ab ↢ bc, ad ↢ bc
Profile 134	bd	bc	acd	ac	c	bc	cd, bc, bcd	a ↢ c, ab ↢ bc, ad ↢ cd
Profile 135	bd	bc	ad	ac	c	bc	cd, acd, bcd	ab ↢ cd
Profile 136	bcd	bc	ad	ac	c	bc	ac, cd, acd	b ↢ c, ab ↢ ac, bd ↢ cd
Profile 137	cd	bc	ad	ac	c	bc	ac, cd, acd	b ↢ c, ab ↢ ac, bd ↢ ac
Profile 138	cd	bc	ad	ac	c	cd	ac, cd, acd	b ↢ c, ab ↢ ac, bd ↢ ac
Profile 139	cd	bc	ad	abc	c	cd	ac, cd, acd	b ↢ c, ab ↢ ac, bd ↢ ac
Profile 140	cd	bc	ad	ab	c	cd	ac, acd, abc	bd ↢ ac
Profile 141	cd	bc	acd	ab	c	cd	ac, bc, abc	d ↢ c, ad ↢ ac, bd ↢ ac
Profile 142	cd	bc	ac	ab	c	cd	ac, bc, abc	d ↢ c, ad ↢ ac, bd ↢ ac
Profile 143	cd	bcd	ac	ab	c	cd	ac, bc, abc	d ↢ c, ad ↢ ac, bd ↢ bc
Profile 144	cd	bd	ac	ab	c	cd	bc, abc, bcd	ad ↢ bc
Profile 145	cd	bd	ac	abc	c	cd	cd, bc, bcd	a ↢ c, ab ↢ bc, ad ↢ bc
Profile 146	cd	bd	ac	bc	c	cd	cd, bc, bcd	a ↢ c, ab ↢ bc, ad ↢ bc
Profile 147	acd	bd	ac	bc	c	cd	cd, bc, bcd	a ↢ c, ab ↢ bc, ad ↢ cd
Profile 148	acd	bd	ac	bc	c	d	cd, bc, bcd	a ↢ c, ab ↢ bc, ad ↢ cd
Profile 149	ad	bd	ac	bc	c	d	cd, acd, bcd	ab ↢ cd

Profile 150	ad	bd	ac	bcd	c	d	cd, acd	b ↢ d, ab ↢ ad, bc ↢ cd
Profile 151	ad	bd	ac	bcd	cd	d	ad, cd, acd	b ↢ d, ab ↢ ad, bc ↢ cd
Profile 152	ad	bd	ac	cd	cd	d	ad, cd, acd	b ↢ d, ab ↢ ad, bc ↢ ad
Profile 153	abd	bd	ac	cd	cd	d	ad, cd, acd	b ↢ d, ab ↢ ad, bc ↢ ad
Profile 154	ab	bd	ac	cd	cd	d	ad, abd, acd	bc ↢ ad
Profile 155	ab	bd	acd	cd	cd	d	ad, bd, abd	c ↢ d, ac ↢ ad, bc ↢ ad
Profile 156	ab	bd	ad	cd	cd	d	ad, bd, abd	c ↢ d, ac ↢ ad, bc ↢ ad
Profile 157	ab	bcd	ad	cd	cd	d	ad, bd, abd	c ↢ d, ac ↢ ad, bc ↢ bd
Profile 158	ab	bc	ad	cd	cd	d	bd, abd, bcd	ac ↢ bd
Profile 159	ab	bc	abd	cd	cd	d	bd, bcd	a ↢ b, ac ↢ bc, ad ↢ bd
Profile 160	ab	bc	bd	cd	cd	d	bd, bcd	a ↢ b, ac ↢ bc, ad ↢ bd
Profile 161	bd	bc	bd	cd	cd	d	bd, cd, bcd	a ↢ b, ab ↢ bc, ac ↢ bc, ad ↢ bd
Profile 162	bd	bc	bd	ac	cd	d	cd, bcd	a ↢ c, ab ↢ bc, ad ↢ cd
Profile 163	bd	bc	bd	ac	acd	d	cd, bcd	a ↢ c, ab ↢ bc, ad ↢ cd
Profile 164	bd	bc	bd	ac	ad	d	cd, acd, bcd	ab ↢ cd
Profile 165	bd	bcd	bd	ac	ad	d	ad, cd, acd	b ↢ d, ab ↢ ad, bc ↢ cd
Profile 166	bd	cd	bd	ac	ad	d	ad, cd, acd	b ↢ d, ab ↢ ad, bc ↢ ad
Profile 167	bd	cd	bd	ac	abd	d	ad, cd, acd	b ↢ d, ab ↢ ad, bc ↢ ad
Profile 168	bd	cd	bd	ac	ab	d	ad, acd, abd	bc ↢ ad
Profile 169	bd	cd	bd	acd	ab	d	ad, bd, abd	c ↢ d, ac ↢ ad, bc ↢ ad
Profile 170	bd	cd	bd	ad	ab	d	ad, bd, abd	c ↢ d, ac ↢ ad, bc ↢ ad
Profile 171	bd	cd	bcd	ad	ab	d	ad, bd, abd	c ↢ d, ac ↢ ad, bc ↢ bd
Profile 172	b	cd	bcd	ad	ab	d	bd, abd	c ↢ d, ac ↢ ad, bc ↢ bd
Profile 173	b	cd	bc	ad	ab	d	bd, abd, bcd	ac ↢ bd
Profile 174	b	cd	bc	abd	ab	d	bd, bcd	a ↢ b, ac ↢ bc, ad ↢ bd
Profile 175	b	cd	bc	bd	ab	d	bd, bcd	a ↢ b, ac ↢ bc, ad ↢ bd
Profile 176	b	cd	bc	bd	ab	bd	bc, bd, bcd	a ↢ b, ac ↢ bc, ad ↢ bc
Profile 177	b	cd	abc	bd	ab	bd	bc, bd, bcd	a ↢ b, ac ↢ bc, ad ↢ bc
Profile 178	b	cd	ac	bd	ab	bd	bc, abc, bcd	ad ↢ bc
Profile 179	b	bcd	ac	bd	ab	bd	ab, bc, abc	d ↢ b, ad ↢ ab, cd ↢ bc
Profile 180	b	bc	ac	bd	ab	bd	ab, bc, abc	d ↢ b, ad ↢ ab, cd ↢ ab
Profile 181	b	bc	ac	bc	ab	bd	ab, bc, abc	d ↢ b, ad ↢ ab, cd ↢ ab
Profile 182	b	bc	ac	bc	abd	bd	ab, bc, abc	d ↢ b, ad ↢ ab, cd ↢ ab
Profile 183	b	bc	ac	bc	ad	bd	ab, abc, abd	cd ↢ ab
Profile 184	b	bc	abc	bc	ad	bd	ab, bd, abd	c ↢ b, ac ↢ ab, cd ↢ ab
Profile 185	b	bc	ab	bc	ad	bd	ab, bd, abd	c ↢ b, ac ↢ ab, cd ↢ ab
Profile 186	b	bc	ab	bc	ad	bcd	ab, bd, abd	c ↢ b, ac ↢ ab, cd ↢ bd
Profile 187	b	bc	ab	bc	ad	cd	bd, abd, bcd	ac ↢ bd
Profile 188	b	bc	ab	bc	abd	cd	bc, bd, bcd	a ↢ b, ac ↢ bc, ad ↢ bd
Profile 189	b	bc	ab	bc	bd	cd	bc, bd, bcd	a ↢ b, ac ↢ bc, ad ↢ bc

Profile 190	*b*	*bc*	*ab*	*abc*	*bd*	*cd*	*bc*, *bd*, *bcd*	*a* ↢ *b*, *ac* ↢ *bc*, *ad* ↢ *bc*
Profile 191	*b*	*c*	*ab*	*abc*	*bd*	*cd*	*bc*, *bd*, *bcd*	*a* ↢ *b*, *ac* ↢ *bc*, *ad* ↢ *bc*
Profile 192	*b*	*c*	*ab*	*ac*	*bd*	*cd*	*bc*, *abc*, *bcd*	*ad* ↢ *bc*

A.3 Assuming $f(\mathcal{A}_1)$ has support *bcd* leads to contradiction.

	A_1	A_2	A_3	A_4	A_5	A_6	possible supports	dominated supports
Profile 191	*b*	*c*	*ab*	*ac*	*bd*	*cd*	*bc*, *abc*, *bcd*	*ad* ↢ *bc*
Profile 192	*b*	*c*	*ab*	*ac*	*bcd*	*cd*	*bc*, *abc*	*d* ↢ *c*, *ad* ↢ *ac*, *bd* ↢ *bc*
Profile 193	*bc*	*c*	*ab*	*ac*	*bcd*	*cd*	*ac*, *bc*, *abc*	*d* ↢ *c*, *ad* ↢ *ac*, *bd* ↢ *bc*
Profile 194	*bc*	*c*	*ab*	*ac*	*bc*	*cd*	*ac*, *bc*, *abc*	*d* ↢ *c*, *ad* ↢ *ac*, *bd* ↢ *ac*
Profile 195	*bc*	*c*	*ab*	*acd*	*bc*	*cd*	*ac*, *bc*, *abc*	*d* ↢ *c*, *ad* ↢ *ac*, *bd* ↢ *ac*
Profile 196	*bc*	*c*	*ab*	*ad*	*bc*	*cd*	*ac*, *acd*, *abc*	*bd* ↢ *ac*
Profile 197	*bc*	*c*	*abc*	*ad*	*bc*	*cd*	*ac*, *cd*, *acd*	*b* ↢ *c*, *ab* ↢ *ac*, *bd* ↢ *ac*
Profile 198	*bc*	*c*	*ac*	*ad*	*bc*	*cd*	*ac*, *cd*, *acd*	*b* ↢ *c*, *ab* ↢ *ac*, *bd* ↢ *ac*
Profile 199	*bc*	*c*	*ac*	*ad*	*bc*	*bcd*	*ac*, *cd*, *acd*	*b* ↢ *c*, *ab* ↢ *ac*, *bd* ↢ *cd*
Profile 200	*bc*	*c*	*ac*	*ad*	*bc*	*bd*	*cd*, *acd*, *bcd*	*ab* ↢ *cd*
Profile 201	*bc*	*c*	*acd*	*ad*	*bc*	*bd*	*cd*, *bcd*	*a* ↢ *d*, *ab* ↢ *bd*, *ac* ↢ *cd*
Profile 202	*bc*	*c*	*cd*	*ad*	*bc*	*bd*	*cd*, *bcd*	*a* ↢ *d*, *ab* ↢ *bd*, *ac* ↢ *cd*
Profile 203	*bc*	*c*	*cd*	*cd*	*bc*	*bd*	*cd*, *bc*, *bcd*	*a* ↢ *b*, *ab* ↢ *bc*, *ac* ↢ *bc*, *ad* ↢ *bc*
Profile 204	*bc*	*c*	*cd*	*cd*	*ab*	*bd*	*bc*, *bcd*	*a* ↢ *b*, *ac* ↢ *bc*, *ad* ↢ *bc*
Profile 205	*abc*	*c*	*cd*	*cd*	*ab*	*bd*	*bc*, *bcd*	*a* ↢ *b*, *ac* ↢ *bc*, *ad* ↢ *bc*
Profile 206	*ac*	*c*	*cd*	*cd*	*ab*	*bd*	*bc*, *abc*, *bcd*	*ad* ↢ *bc*
Profile 207	*ac*	*c*	*cd*	*cd*	*ab*	*bcd*	*ac*, *bc*, *abc*	*d* ↢ *c*, *ad* ↢ *ac*, *bd* ↢ *bc*
Profile 208	*ac*	*c*	*cd*	*cd*	*ab*	*bc*	*ac*, *bc*, *abc*	*d* ↢ *c*, *ad* ↢ *ac*, *bd* ↢ *ac*
Profile 209	*acd*	*c*	*cd*	*cd*	*ab*	*bc*	*ac*, *bc*, *abc*	*d* ↢ *c*, *ad* ↢ *ac*, *bd* ↢ *ac*
Profile 210	*ad*	*c*	*cd*	*cd*	*ab*	*bc*	*ac*, *abc*, *acd*	*bd* ↢ *ac*
Profile 211	*ad*	*c*	*cd*	*cd*	*abc*	*bc*	*ac*, *cd*, *acd*	*b* ↢ *c*, *ab* ↢ *ac*, *bd* ↢ *ac*
Profile 212	*ad*	*c*	*cd*	*cd*	*ac*	*bc*	*ac*, *cd*, *acd*	*b* ↢ *c*, *ab* ↢ *ac*, *bd* ↢ *ac*
Profile 213	*ad*	*c*	*bcd*	*cd*	*ac*	*bc*	*ac*, *cd*, *acd*	*b* ↢ *c*, *ab* ↢ *ac*, *bd* ↢ *cd*
Profile 214	*ad*	*c*	*bcd*	*d*	*ac*	*bc*	*cd*, *acd*	*b* ↢ *c*, *ab* ↢ *ac*, *bd* ↢ *cd*
Profile 215	*ad*	*c*	*bd*	*d*	*ac*	*bc*	*cd*, *acd*, *bcd*	*ab* ↢ *cd*
Profile 216	*ad*	*c*	*bd*	*d*	*acd*	*bc*	*cd*, *bcd*	*a* ↢ *d*, *ab* ↢ *bd*, *ac* ↢ *cd*
Profile 217	*ad*	*c*	*bd*	*d*	*cd*	*bc*	*cd*, *bcd*	*a* ↢ *d*, *ab* ↢ *bd*, *ac* ↢ *cd*
Profile 218	*ad*	*cd*	*bd*	*d*	*cd*	*bc*	*bd*, *cd*, *bcd*	*a* ↢ *d*, *ab* ↢ *bd*, *ac* ↢ *bd*
Profile 219	*ad*	*cd*	*abd*	*d*	*cd*	*bc*	*bd*, *cd*, *bcd*	*a* ↢ *d*, *ab* ↢ *bd*, *ac* ↢ *bd*
Profile 220	*ad*	*cd*	*ab*	*d*	*cd*	*bc*	*bd*, *abd*, *bcd*	*ac* ↢ *bd*
Profile 221	*ad*	*cd*	*ab*	*d*	*cd*	*bcd*	*ad*, *bd*, *abd*	*c* ↢ *d*, *ac* ↢ *ad*, *bc* ↢ *bd*
Profile 222	*ad*	*cd*	*ab*	*d*	*cd*	*bd*	*ad*, *bd*, *abd*	*c* ↢ *d*, *ac* ↢ *ad*, *bc* ↢ *ad*
Profile 223	*ad*	*cd*	*ab*	*d*	*bd*	*bd*	*ad*, *bd*, *abd*	*c* ↢ *d*, *ac* ↢ *ad*, *bc* ↢ *ad*

Profile 224	acd	cd	ab	d	bd	bd	ad, bd, abd	c ↢ d, ac ↢ ad, bc ↢ ad
Profile 225	ac	cd	ab	d	bd	bd	ad, abd, acd	bc ↢ ad
Profile 226	ac	cd	abd	d	bd	bd	ad, cd, acd	b ↢ d, ab ↢ ad, bc ↢ ad
Profile 227	ac	cd	ad	d	bd	bd	ad, cd, acd	b ↢ d, ab ↢ ad, bc ↢ ad
Profile 228	ac	bcd	ad	d	bd	bd	ad, cd, acd	b ↢ d, ab ↢ ad, bc ↢ cd
Profile 229	ac	bc	ad	d	bd	bd	cd, acd, bcd	ab ↢ cd
Profile 230	acd	bc	ad	d	bd	bd	bd, cd, bcd	a ↢ d, ab ↢ bd, ac ↢ cd
Profile 231	cd	bc	ad	d	bd	bd	bd, cd, bcd	a ↢ d, ab ↢ bd, ac ↢ bd
Profile 232	cd	bc	ad	d	bd	abd	bd, cd, bcd	a ↢ d, ab ↢ bd, ac ↢ bd
Profile 233	cd	bc	ad	d	b	abd	bd, cd, bcd	a ↢ d, ab ↢ bd, ac ↢ bd
Profile 234	cd	bc	ad	d	b	ab	bd, abd, bcd	ac ↢ bd
Profile 235	bcd	bc	ad	d	b	ab	bd, abd	c ↢ b, ac ↢ ab, cd ↢ bd
Profile 236	bcd	bc	ad	bd	b	ab	ab, bd, abd	c ↢ b, ac ↢ ab, cd ↢ bd
Profile 237	bd	bc	ad	bd	b	ab	ab, bd, abd	c ↢ b, ac ↢ ab, cd ↢ ab
Profile 238	bd	bc	ad	bd	b	abc	ab, bd, abd	c ↢ b, ac ↢ ab, cd ↢ ab
Profile 239	bd	bc	ad	bd	b	ac	ab, abc, abd	cd ↢ ab
Profile 240	bd	bc	abd	bd	b	ac	ab, bc, abc	d ↢ b, ad ↢ ab, cd ↢ ab
Profile 241	bd	bc	ab	bd	b	ac	ab, bc, abc	d ↢ b, ad ↢ ab, cd ↢ ab
Profile 242	bd	bcd	ab	bd	b	ac	ab, bc, abc	d ↢ b, ad ↢ ab, cd ↢ bc
Profile 243	bd	cd	ab	bd	b	ac	bc, abc, bcd	ad ↢ bc
Profile 244	bd	cd	abc	bd	b	ac	bc, bcd	a ↢ c, ab ↢ bc, ad ↢ bc
Profile 245	bd	cd	bc	bd	b	ac	bc, bcd	a ↢ c, ab ↢ bc, ad ↢ bc
Profile 246	bd	cd	bc	bd	b	bc	bc, bd, bcd	a ↢ b, ab ↢ bc, ac ↢ bc, ad ↢ bc
Profile 247	ad	cd	bc	bd	b	bc	bd, bcd	a ↢ d, ab ↢ bd, ac ↢ bd
Profile 248	ad	cd	bc	abd	b	bc	bd, bcd	a ↢ d, ab ↢ bd, ac ↢ bd
Profile 249	ad	cd	bc	ab	b	bc	bd, abd, bcd	ac ↢ bd
Profile 250	ad	bcd	bc	ab	b	bc	ab, bd, abd	c ↢ b, ac ↢ ab, cd ↢ bd
Profile 251	ad	bd	bc	ab	b	bc	ab, bd, abd	c ↢ b, ac ↢ ab, cd ↢ ab
Profile 252	ad	bd	bc	abc	b	bc	ab, bd, abd	c ↢ b, ac ↢ ab, cd ↢ ab
Profile 253	ad	bd	bc	ac	b	bc	ab, abd, abc	cd ↢ ab
Profile 254	abd	bd	bc	ac	b	bc	ab, bc, abc	d ↢ b, ad ↢ ab, cd ↢ ab
Profile 255	ab	bd	bc	ac	b	bc	ab, bc, abc	d ↢ b, ad ↢ ab, cd ↢ ab
Profile 256	ab	bd	bcd	ac	b	bc	ab, bc, abc	d ↢ b, ad ↢ ab, cd ↢ bc
Profile 257	ab	bd	bcd	ac	b	c	bc, abc	d ↢ b, ad ↢ ab, cd ↢ bc
Profile 258	ab	bd	cd	ac	b	c	bc, abc, bcd	ad ↢ bc
Profile 259	abc	bd	cd	ac	b	c	bc, bcd	a ↢ c, ab ↢ bc, ad ↢ bc
Profile 260	bc	bd	cd	ac	b	c	bc, bcd	a ↢ c, ab ↢ bc, ad ↢ bc
Profile 261	bc	bd	cd	ac	bc	c	cd, bc, bcd	a ↢ c, ab ↢ bc, ad ↢ bc
Profile 262	bc	bd	acd	ac	bc	c	cd, bc, bcd	a ↢ c, ab ↢ bc, ad ↢ cd
Profile 263	bc	bd	ad	ac	bc	c	cd, acd, bcd	ab ↢ cd

Profile 264	bc	bcd	ad	ac	bc	c	ac, cd, acd	$b \leftarrowtail c,\ ab \leftarrowtail ac,\ bd \leftarrowtail cd$
Profile 265	bc	cd	ad	ac	bc	c	ac, cd, acd	$b \leftarrowtail c,\ ab \leftarrowtail ac,\ bd \leftarrowtail ac$
Profile 266	cd	cd	ad	ac	bc	c	ac, cd, acd	$b \leftarrowtail c,\ ab \leftarrowtail ac,\ bd \leftarrowtail ac$
Profile 267	cd	cd	ad	abc	bc	c	ac, cd, acd	$b \leftarrowtail c,\ ab \leftarrowtail ac,\ bd \leftarrowtail ac$
Profile 268	cd	cd	ad	ab	bc	c	ac, acd, abc	$bd \leftarrowtail ac$
Profile 269	cd	cd	acd	ab	bc	c	ac, bc, abc	$d \leftarrowtail c,\ ad \leftarrowtail ac,\ bd \leftarrowtail ac$
Profile 270	cd	cd	ac	ab	bc	c	ac, bc, abc	$d \leftarrowtail c,\ ad \leftarrowtail ac,\ bd \leftarrowtail ac$
Profile 271	cd	cd	ac	ab	bcd	c	ac, bc, abc	$d \leftarrowtail c,\ ad \leftarrowtail ac,\ bd \leftarrowtail bc$
Profile 272	cd	cd	ac	ab	bd	c	bc, abc, bcd	$ad \leftarrowtail bc$
Profile 273	cd	cd	ac	abc	bd	c	cd, bc, bcd	$a \leftarrowtail c,\ ab \leftarrowtail bc,\ ad \leftarrowtail bc$
Profile 274	cd	cd	ac	bc	bd	c	cd, bc, bcd	$a \leftarrowtail c,\ ab \leftarrowtail bc,\ ad \leftarrowtail bc$
Profile 275	cd	acd	ac	bc	bd	c	cd, bc, bcd	$a \leftarrowtail c,\ ab \leftarrowtail bc,\ ad \leftarrowtail cd$
Profile 276	d	acd	ac	bc	bd	c	cd, bc, bcd	$a \leftarrowtail c,\ ab \leftarrowtail bc,\ ad \leftarrowtail cd$
Profile 277	d	ad	ac	bc	bd	c	cd, acd, bcd	$ab \leftarrowtail cd$
Profile 278	d	ad	ac	bcd	bd	c	cd, acd	$b \leftarrowtail d,\ ab \leftarrowtail ad,\ bc \leftarrowtail cd$
Profile 279	d	ad	ac	bcd	bd	cd	ad, cd, acd	$b \leftarrowtail d,\ ab \leftarrowtail ad,\ bc \leftarrowtail cd$
Profile 280	d	ad	ac	cd	bd	cd	ad, cd, acd	$b \leftarrowtail d,\ ab \leftarrowtail ad,\ bc \leftarrowtail ad$
Profile 281	d	abd	ac	cd	bd	cd	ad, cd, acd	$b \leftarrowtail d,\ ab \leftarrowtail ad,\ bc \leftarrowtail ad$
Profile 282	d	ab	ac	cd	bd	cd	ad, abd, acd	$bc \leftarrowtail ad$
Profile 283	d	ab	acd	cd	bd	cd	ad, bd, abd	$c \leftarrowtail d,\ ac \leftarrowtail ad,\ bc \leftarrowtail ad$
Profile 284	d	ab	ad	cd	bd	cd	ad, bd, abd	$c \leftarrowtail d,\ ac \leftarrowtail ad,\ bc \leftarrowtail ad$
Profile 285	d	ab	ad	cd	bcd	cd	ad, bd, abd	$c \leftarrowtail d,\ ac \leftarrowtail ad,\ bc \leftarrowtail bd$
Profile 286	d	ab	ad	cd	bc	cd	bd, abd, bcd	$ac \leftarrowtail bd$
Profile 287	d	ab	abd	cd	bc	cd	bd, bcd	$a \leftarrowtail b,\ ac \leftarrowtail bc,\ ad \leftarrowtail bd$
Profile 288	d	ab	bd	cd	bc	cd	bd, bcd	$a \leftarrowtail b,\ ac \leftarrowtail bc,\ ad \leftarrowtail bd$
Profile 289	d	bd	bd	cd	bc	cd	bd, cd, bcd	$a \leftarrowtail b,\ ab \leftarrowtail bc,\ ac \leftarrowtail bc,\ ad \leftarrowtail bd$
Profile 290	d	bd	bd	ac	bc	cd	cd, bcd	$a \leftarrowtail c,\ ab \leftarrowtail bc,\ ad \leftarrowtail cd$
Profile 291	d	bd	bd	ac	bc	acd	cd, bcd	$a \leftarrowtail c,\ ab \leftarrowtail bc,\ ad \leftarrowtail cd$
Profile 292	d	bd	bd	ac	bc	ad	cd, acd, bcd	$ab \leftarrowtail cd$
Profile 293	d	bd	bd	ac	bcd	ad	ad, cd, acd	$b \leftarrowtail d,\ ab \leftarrowtail ad,\ bc \leftarrowtail cd$
Profile 294	d	bd	bd	ac	cd	ad	ad, cd, acd	$b \leftarrowtail d,\ ab \leftarrowtail ad,\ bc \leftarrowtail ad$
Profile 295	d	bd	bd	ac	cd	abd	ad, cd, acd	$b \leftarrowtail d,\ ab \leftarrowtail ad,\ bc \leftarrowtail ad$
Profile 296	d	bd	bd	ac	cd	ab	ad, acd, abd	$bc \leftarrowtail ad$
Profile 297	d	bd	bd	acd	cd	ab	ad, bd, abd	$c \leftarrowtail d,\ ac \leftarrowtail ad,\ bc \leftarrowtail ad$
Profile 298	d	bd	bd	ad	cd	ab	ad, bd, abd	$c \leftarrowtail d,\ ac \leftarrowtail ad,\ bc \leftarrowtail ad$
Profile 299	d	bcd	bd	ad	cd	ab	ad, bd, abd	$c \leftarrowtail d,\ ac \leftarrowtail ad,\ bc \leftarrowtail bd$
Profile 300	d	bcd	b	ad	cd	ab	bd, abd	$c \leftarrowtail d,\ ac \leftarrowtail ad,\ bc \leftarrowtail bd$
Profile 301	d	bc	b	ad	cd	ab	bd, abd, bcd	$ac \leftarrowtail bd$
Profile 302	d	bc	b	abd	cd	ab	bd, bcd	$a \leftarrowtail b,\ ac \leftarrowtail bc,\ ad \leftarrowtail bd$
Profile 303	d	bc	b	bd	cd	ab	bd, bcd	$a \leftarrowtail b,\ ac \leftarrowtail bc,\ ad \leftarrowtail bd$

Profile 304	*bd*	*bc*	*b*	*bd*	*cd*	*ab*	*bc*, *bd*, *bcd*	*a* ↢ *b*, *ac* ↢ *bc*, *ad* ↢ *bc*
Profile 305	*bd*	*abc*	*b*	*bd*	*cd*	*ab*	*bc*, *bd*, *bcd*	*a* ↢ *b*, *ac* ↢ *bc*, *ad* ↢ *bc*
Profile 306	*bd*	*ac*	*b*	*bd*	*cd*	*ab*	*bc*, *abc*, *bcd*	*ad* ↢ *bc*
Profile 307	*bd*	*ac*	*b*	*bd*	*bcd*	*ab*	*ab*, *bc*, *abc*	*d* ↢ *b*, *ad* ↢ *ab*, *cd* ↢ *bc*
Profile 308	*bd*	*ac*	*b*	*bd*	*bc*	*ab*	*ab*, *bc*, *abc*	*d* ↢ *b*, *ad* ↢ *ab*, *cd* ↢ *ab*
Profile 309	*bd*	*ac*	*b*	*bc*	*bc*	*ab*	*ab*, *bc*, *abc*	*d* ↢ *b*, *ad* ↢ *ab*, *cd* ↢ *ab*
Profile 310	*bd*	*ac*	*b*	*bc*	*bc*	*abd*	*ab*, *bc*, *abc*	*d* ↢ *b*, *ad* ↢ *ab*, *cd* ↢ *ab*
Profile 311	*bd*	*ac*	*b*	*bc*	*bc*	*ad*	*ab*, *abc*, *abd*	*cd* ↢ *ab*
Profile 312	*bd*	*abc*	*b*	*bc*	*bc*	*ad*	*ab*, *bd*, *abd*	*c* ↢ *b*, *ac* ↢ *ab*, *cd* ↢ *ab*
Profile 313	*bd*	*ab*	*b*	*bc*	*bc*	*ad*	*ab*, *bd*, *abd*	*c* ↢ *b*, *ac* ↢ *ab*, *cd* ↢ *ab*
Profile 314	*bcd*	*ab*	*b*	*bc*	*bc*	*ad*	*ab*, *bd*, *abd*	*c* ↢ *b*, *ac* ↢ *ab*, *cd* ↢ *bd*
Profile 315	*cd*	*ab*	*b*	*bc*	*bc*	*ad*	*bd*, *abd*, *bcd*	*ac* ↢ *bd*
Profile 316	*cd*	*ab*	*b*	*bc*	*bc*	*abd*	*bc*, *bd*, *bcd*	*a* ↢ *b*, *ac* ↢ *bc*, *ad* ↢ *bd*
Profile 317	*cd*	*ab*	*b*	*bc*	*bc*	*bd*	*bc*, *bd*, *bcd*	*a* ↢ *b*, *ac* ↢ *bc*, *ad* ↢ *bc*
Profile 318	*cd*	*ab*	*b*	*abc*	*bc*	*bd*	*bc*, *bd*, *bcd*	*a* ↢ *b*, *ac* ↢ *bc*, *ad* ↢ *bc*
Profile 319	*cd*	*ab*	*b*	*abc*	*c*	*bd*	*bc*, *bd*, *bcd*	*a* ↢ *b*, *ac* ↢ *bc*, *ad* ↢ *bc*
Profile 320	*cd*	*ab*	*b*	*ac*	*c*	*bd*	*bc*, *abc*, *bcd*	*ad* ↢ *bc*
Profile 321	*cd*	*ab*	*b*	*ac*	*c*	*bcd*	*bc*, *abc*	*d* ↢ *c*, *ad* ↢ *ac*, *bd* ↢ *bc*
Profile 322	*cd*	*ab*	*bc*	*ac*	*c*	*bcd*	*ac*, *bc*, *abc*	*d* ↢ *c*, *ad* ↢ *ac*, *bd* ↢ *bc*
Profile 323	*cd*	*ab*	*bc*	*ac*	*c*	*bc*	*ac*, *bc*, *abc*	*d* ↢ *c*, *ad* ↢ *ac*, *bd* ↢ *ac*
Profile 324	*cd*	*ab*	*bc*	*acd*	*c*	*bc*	*ac*, *bc*, *abc*	*d* ↢ *c*, *ad* ↢ *ac*, *bd* ↢ *ac*
Profile 325	*cd*	*ab*	*bc*	*ad*	*c*	*bc*	*ac*, *acd*, *abc*	*bd* ↢ *ac*
Profile 326	*cd*	*abc*	*bc*	*ad*	*c*	*bc*	*ac*, *cd*, *acd*	*b* ↢ *c*, *ab* ↢ *ac*, *bd* ↢ *ac*
Profile 327	*cd*	*ac*	*bc*	*ad*	*c*	*bc*	*ac*, *cd*, *acd*	*b* ↢ *c*, *ab* ↢ *ac*, *bd* ↢ *ac*
Profile 328	*bcd*	*ac*	*bc*	*ad*	*c*	*bc*	*ac*, *cd*, *acd*	*b* ↢ *c*, *ab* ↢ *ac*, *bd* ↢ *cd*
Profile 329	*bd*	*ac*	*bc*	*ad*	*c*	*bc*	*cd*, *acd*, *bcd*	*ab* ↢ *cd*
Profile 330	*bd*	*acd*	*bc*	*ad*	*c*	*bc*	*cd*, *bcd*	*a* ↢ *d*, *ab* ↢ *bd*, *ac* ↢ *cd*
Profile 331	*bd*	*cd*	*bc*	*ad*	*c*	*bc*	*cd*, *bcd*	*a* ↢ *d*, *ab* ↢ *bd*, *ac* ↢ *cd*
Profile 332	*bd*	*cd*	*bc*	*cd*	*c*	*bc*	*cd*, *bc*, *bcd*	*a* ↢ *b*, *ab* ↢ *bc*, *ac* ↢ *bc*, *ad* ↢ *bc*
Profile 333	*bd*	*cd*	*bc*	*cd*	*c*	*ab*	*bc*, *bcd*	*a* ↢ *b*, *ac* ↢ *bc*, *ad* ↢ *bc*
Profile 334	*bd*	*cd*	*abc*	*cd*	*c*	*ab*	*bc*, *bcd*	*a* ↢ *b*, *ac* ↢ *bc*, *ad* ↢ *bc*
Profile 335	*bd*	*cd*	*ac*	*cd*	*c*	*ab*	*bc*, *abc*, *bcd*	*ad* ↢ *bc*
Profile 336	*bcd*	*cd*	*ac*	*cd*	*c*	*ab*	*ac*, *bc*, *abc*	*d* ↢ *c*, *ad* ↢ *ac*, *bd* ↢ *bc*
Profile 337	*bc*	*cd*	*ac*	*cd*	*c*	*ab*	*ac*, *bc*, *abc*	*d* ↢ *c*, *ad* ↢ *ac*, *bd* ↢ *ac*
Profile 338	*bc*	*cd*	*acd*	*cd*	*c*	*ab*	*ac*, *bc*, *abc*	*d* ↢ *c*, *ad* ↢ *ac*, *bd* ↢ *ac*
Profile 339	*bc*	*cd*	*ad*	*cd*	*c*	*ab*	*ac*, *abc*, *acd*	*bd* ↢ *ac*
Profile 340	*bc*	*cd*	*ad*	*cd*	*c*	*abc*	*ac*, *cd*, *acd*	*b* ↢ *c*, *ab* ↢ *ac*, *bd* ↢ *ac*
Profile 341	*bc*	*cd*	*ad*	*cd*	*c*	*ac*	*ac*, *cd*, *acd*	*b* ↢ *c*, *ab* ↢ *ac*, *bd* ↢ *ac*
Profile 342	*bc*	*bcd*	*ad*	*cd*	*c*	*ac*	*ac*, *cd*, *acd*	*b* ↢ *c*, *ab* ↢ *ac*, *bd* ↢ *cd*
Profile 343	*bc*	*bcd*	*ad*	*d*	*c*	*ac*	*cd*, *acd*	*b* ↢ *c*, *ab* ↢ *ac*, *bd* ↢ *cd*

Profile 344	*bc*	*bd*	*ad*	*d*	*c*	*ac*	*cd*, *acd*, *bcd*	*ab* ↢ *cd*
Profile 345	*bc*	*bd*	*ad*	*d*	*c*	*acd*	*cd*, *bcd*	*a* ↢ *d*, *ab* ↢ *bd*, *ac* ↢ *cd*
Profile 346	*bc*	*bd*	*ad*	*d*	*c*	*cd*	*cd*, *bcd*	*a* ↢ *d*, *ab* ↢ *bd*, *ac* ↢ *cd*
Profile 347	*bc*	*bd*	*ad*	*d*	*cd*	*cd*	*bd*, *cd*, *bcd*	*a* ↢ *d*, *ab* ↢ *bd*, *ac* ↢ *bd*
Profile 348	*bc*	*abd*	*ad*	*d*	*cd*	*cd*	*bd*, *cd*, *bcd*	*a* ↢ *d*, *ab* ↢ *bd*, *ac* ↢ *bd*
Profile 349	*bc*	*ab*	*ad*	*d*	*cd*	*cd*	*bd*, *abd*, *bcd*	*ac* ↢ *bd*
Profile 350	*bcd*	*ab*	*ad*	*d*	*cd*	*cd*	*ad*, *bd*, *abd*	*c* ↢ *d*, *ac* ↢ *ad*, *bc* ↢ *bd*
Profile 351	*bd*	*ab*	*ad*	*d*	*cd*	*cd*	*ad*, *bd*, *abd*	*c* ↢ *d*, *ac* ↢ *ad*, *bc* ↢ *ad*
Profile 352	*bd*	*ab*	*ad*	*d*	*cd*	*bd*	*ad*, *bd*, *abd*	*c* ↢ *d*, *ac* ↢ *ad*, *bc* ↢ *ad*
Profile 353	*bd*	*ab*	*acd*	*d*	*cd*	*bd*	*ad*, *bd*, *abd*	*c* ↢ *d*, *ac* ↢ *ad*, *bc* ↢ *ad*
Profile 354	*bd*	*ab*	*ac*	*d*	*cd*	*bd*	*ad*, *abd*, *acd*	*bc* ↢ *ad*
Profile 355	*bd*	*abd*	*ac*	*d*	*cd*	*bd*	*ad*, *cd*, *acd*	*b* ↢ *d*, *ab* ↢ *ad*, *bc* ↢ *ad*
Profile 356	*bd*	*ad*	*ac*	*d*	*cd*	*bd*	*ad*, *cd*, *acd*	*b* ↢ *d*, *ab* ↢ *ad*, *bc* ↢ *ad*
Profile 357	*bd*	*ad*	*ac*	*d*	*bcd*	*bd*	*ad*, *cd*, *acd*	*b* ↢ *d*, *ab* ↢ *ad*, *bc* ↢ *cd*
Profile 358	*bd*	*ad*	*ac*	*d*	*bc*	*bd*	*cd*, *acd*, *bcd*	*ab* ↢ *cd*
Profile 359	*bd*	*ad*	*acd*	*d*	*bc*	*bd*	*bd*, *cd*, *bcd*	*a* ↢ *d*, *ab* ↢ *bd*, *ac* ↢ *cd*
Profile 360	*bd*	*ad*	*cd*	*d*	*bc*	*bd*	*bd*, *cd*, *bcd*	*a* ↢ *d*, *ab* ↢ *bd*, *ac* ↢ *bd*
Profile 361	*bd*	*ad*	*cd*	*d*	*bc*	*abd*	*bd*, *cd*, *bcd*	*a* ↢ *d*, *ab* ↢ *bd*, *ac* ↢ *bd*
Profile 362	*b*	*ad*	*cd*	*d*	*bc*	*abd*	*bd*, *cd*, *bcd*	*a* ↢ *d*, *ab* ↢ *bd*, *ac* ↢ *bd*
Profile 363	*b*	*ad*	*cd*	*d*	*bc*	*ab*	*bd*, *abd*, *bcd*	*ac* ↢ *bd*
Profile 364	*b*	*ad*	*bcd*	*d*	*bc*	*ab*	*bd*, *abd*	*c* ↢ *b*, *ac* ↢ *ab*, *cd* ↢ *bd*
Profile 365	*b*	*ad*	*bcd*	*bd*	*bc*	*ab*	*ab*, *bd*, *abd*	*c* ↢ *b*, *ac* ↢ *ab*, *cd* ↢ *bd*
Profile 366	*b*	*ad*	*bd*	*bd*	*bc*	*ab*	*ab*, *bd*, *abd*	*c* ↢ *b*, *ac* ↢ *ab*, *cd* ↢ *ab*
Profile 367	*b*	*ad*	*bd*	*bd*	*bc*	*abc*	*ab*, *bd*, *abd*	*c* ↢ *b*, *ac* ↢ *ab*, *cd* ↢ *ab*
Profile 368	*b*	*ad*	*bd*	*bd*	*bc*	*ac*	*ab*, *abc*, *abd*	*cd* ↢ *ab*
Profile 369	*b*	*abd*	*bd*	*bd*	*bc*	*ac*	*ab*, *bc*, *abc*	*d* ↢ *b*, *ad* ↢ *ab*, *cd* ↢ *ab*
Profile 370	*b*	*ab*	*bd*	*bd*	*bc*	*ac*	*ab*, *bc*, *abc*	*d* ↢ *b*, *ad* ↢ *ab*, *cd* ↢ *ab*
Profile 371	*b*	*ab*	*bd*	*bd*	*bcd*	*ac*	*ab*, *bc*, *abc*	*d* ↢ *b*, *ad* ↢ *ab*, *cd* ↢ *bc*
Profile 372	*b*	*ab*	*bd*	*bd*	*cd*	*ac*	*bc*, *abc*, *bcd*	*ad* ↢ *bc*
Profile 373	*b*	*abc*	*bd*	*bd*	*cd*	*ac*	*bc*, *bcd*	*a* ↢ *c*, *ab* ↢ *bc*, *ad* ↢ *bc*
Profile 374	*b*	*bc*	*bd*	*bd*	*cd*	*ac*	*bc*, *bcd*	*a* ↢ *c*, *ab* ↢ *bc*, *ad* ↢ *bc*
Profile 375	*b*	*bc*	*bd*	*bd*	*cd*	*bc*	*bc*, *bd*, *bcd*	*a* ↢ *b*, *ab* ↢ *bc*, *ac* ↢ *bc*, *ad* ↢ *bc*
Profile 376	*b*	*bc*	*ad*	*bd*	*cd*	*bc*	*bd*, *bcd*	*a* ↢ *d*, *ab* ↢ *bd*, *ac* ↢ *bd*
Profile 377	*b*	*bc*	*ad*	*abd*	*cd*	*bc*	*bd*, *bcd*	*a* ↢ *d*, *ab* ↢ *bd*, *ac* ↢ *bd*
Profile 378	*b*	*bc*	*ad*	*ab*	*cd*	*bc*	*bd*, *abd*, *bcd*	*ac* ↢ *bd*
Profile 379	*b*	*bc*	*ad*	*ab*	*bcd*	*bc*	*ab*, *bd*, *abd*	*c* ↢ *b*, *ac* ↢ *ab*, *cd* ↢ *bd*
Profile 380	*b*	*bc*	*ad*	*ab*	*bd*	*bc*	*ab*, *bd*, *abd*	*c* ↢ *b*, *ac* ↢ *ab*, *cd* ↢ *ab*
Profile 381	*b*	*bc*	*ad*	*abc*	*bd*	*bc*	*ab*, *bd*, *abd*	*c* ↢ *b*, *ac* ↢ *ab*, *cd* ↢ *ab*
Profile 382	*b*	*bc*	*ad*	*ac*	*bd*	*bc*	*ab*, *abd*, *abc*	*cd* ↢ *ab*
Profile 383	*b*	*bc*	*abd*	*ac*	*bd*	*bc*	*ab*, *bc*, *abc*	*d* ↢ *b*, *ad* ↢ *ab*, *cd* ↢ *ab*

Profile 384	b	bc	ab	ac	bd	bc	$\underline{ab}, bc, abc$	$d \leftarrowtail b,\ ad \leftarrowtail ab,\ cd \leftarrowtail ab$
Profile 385	b	bc	ab	ac	bd	bcd	$\underline{ab}, bc, \underline{abc}$	$d \leftarrowtail b,\ ad \leftarrowtail ab,\ cd \leftarrowtail bc$
Profile 386	b	c	ab	ac	bd	bcd	$bc, \underline{abc}$	$d \leftarrowtail b,\ ad \leftarrowtail ab,\ cd \leftarrowtail bc$
Profile 387	b	c	ab	ac	bd	cd	$bc, \underline{abc}, bcd$	$ad \leftarrowtail bc$

B PROOF OF THEOREM 3

Theorem. No anonymous and neutral distribution rule satisfies efficiency, subset strategyproofness, and positive share when $m \geq 5$ and $n \geq 5$.

Proof. We prove the result for $m = 5$ and $n = 5$, and it can be adapted to larger values as before.

Assume that f is a distribution rule satisfying efficiency and positive share. Now consider the following profile $\mathcal{A}$ (with uniform contributions):

$$\mathcal{A} = (\{a\}, \{a, b, c\}, \{a, b, d\}, \{a, c, e\}, \{d, e\}).$$

Let $\delta = f(\mathcal{A})$. Since f is efficient, we must have $\delta(b) = \delta(c) = 0$, as both b and c are dominated by a. Since the profile is symmetric under the permutation $\sigma = (b\ c)(d\ e)$, we have $\delta(b) = \delta(c)$ and $\delta(d) = \delta(e)$ since f is anonymous and neutral. By positive share for agent 5, we have $\delta(d) = \delta(e) > 0$. Hence $u_4(\delta) < C$ since a positive amount is spent on d, which agent 4 does not approve.

Now, suppose that agent 4 pretends not to approve a, leading to the profile

$$\mathcal{A}' = (\{a\}, \{a, b, c\}, \{a, b, d\}, \{c, e\}, \{d, e\}).$$

Let $\delta' = f(\mathcal{A}')$ be the distribution now returned by the rule. Again, by efficiency, we must have $\delta'(b) = 0$ because b is dominated by a. Now, c and d are symmetric projects in $\mathcal{A}'$, and thus we have $\delta'(c) = \delta'(d)$ by anonymity and neutrality of f. If $\delta'(c) = \delta'(d)$ is positive, say equal to $\epsilon > 0$, then δ' is dominated by the distribution obtained from δ' by moving ϵ from c to a and ϵ from d to e. This contradicts efficiency of f. Thus $\delta'(c) = \delta'(d) = 0$ and the entire endowment is distributed between projects a and e, and so $u_4(\delta') = C$. Thus, agent 4 has successfully manipulated f by reporting a subset of her true approval set. □