A general elicitation-free protocol for allocating indivisible goods

#Tracking number 1041

February 15, 2011

Abstract

A benevolent central authority has to allocate a set of indivisible to a set of agents whose preferences it is totally ignorant of. We consider the process of allocating objects one after the other by designating an agent and asking her to pick one of the objects among those that remain. The problem consists in choosing the "best" sequence of agents, according to some optimality criterion. We assume that agents have additive preferences over objects. The choice of an optimality criterion depends on three parameters: how utilities of objects are related to their ranking in an agent's preference relation; how the preferences of different agents are correlated; and how social welfare is defined from the agents' utilities. We address the computation of a sequence maximizing expected social welfare under several assumptions. We also address strategical issues.

1 Introduction

Approaches to fair division, and more generally resource allocation, can be classified according to four dichotomies [Chevaleyre *et al.*, 2006]: (a) divisible vs. indivisible objects; (b) centralized vs. decentralized approaches; (c) revenue efficiency vs. fairness criteria; (d) allowing money transfers or not. (a), (c) and (d) are self-explanatory. As for (b), in *centralized* approaches, the agents communicate their preferences to some central authority, which computes the optimal allocation; in *decentralized* approaches, agents interact with each other (possibly with the help of a central authority), through actions that reveal only a part of their preferences. A typical class of decentralized approaches of this type is the class of *cake cutting procedures* (*e.g.*, [Robertson and Webb, 1998; Brams and Taylor, 1996]), which are typically designed for the allocation of divisible goods. Centralized approaches have two drawbacks: (a) the elicitation process and the winner determination algorithm can be very expensive; (b) the agents have to reveal their full preferences, which they might be reluctant to do.

Although many centralized approaches to allocating indivisible goods have been proposed, decentralized approaches are much less frequent, up to a few exceptions such as [Brams *et al.*, 2011] who adapt a cake-cutting protocol to the allocation of indivisible goods, and [Chevaleyre *et al.*, 2010], who study distributed mechanisms for allocating indivisible objects based agents negotiating so as to exchange objects.

Here we study a much simpler decentralized protocol for allocating indivisible objects, which is used in a variety of daily situations. We have p indivisible objects to allocate to n agents. The central authority defines a sequence of agents of length p. Every time an agent is designated, she picks one object out of those that remain. For instance, if $n = 3$ and $p = 5$, the sequence 12332 means that agent 1 picks an object first; then 2 picks an object; then 3 picks two objects; and 2 takes the last object. The central authority has to find the best sequence, according to some criterion. For instance, if we want to be fair, $\pi = 12332$ seems better than $\pi' = 12321$, because in π, agent 1, who receives only one object, is compensated by the fact that he ends up with his preferred object. This process is arguably very natural. Brams and Taylor [Brams and Taylor, 2000] give it some attention, by studying particular sequences, namely *strict alternation*, where two agents pick objects in alternation, and *balanced alternation* (for two agents) consisting of sequences of the form 1221, 12212112 etc. However they do not justify these sequences by optimality arguments. In fact, we do not know of any work on the definition and computation of *optimal* sequences[1].

To make this formal, we need to define a (generic) model, as follows: (i) agents have additive utilities (the value of a subset of objects is the sum of the values of its elements); (ii) we have a *scoring* function mapping the rank of an object in the preference relation of an agent to its utility value, and this function is the same for all agents; (iii) we have a probability distribution on the possible collections of rankings (or *profiles*); we focus on two prototypical models: full independence (all profiles are equally probable, hence the rankings of two different agents are independent), and full correlation (agents have identical preferences). (i), (ii) and (iii) suffice to determine the expected utility of an agent for a given sequence. Finally, (iv) we have a social welfare function F aggregating the utilities of the agents; we will focus on $F = +$ (utilitarianism) and $F = \min$ (egalitarianism). We then look for a sequence optimizing the expected social welfare.

The general model is defined in Section 2. In Section 3 we address the computation of the optimal sequence under various instances of the model. In Section 4 we consider strategical issues. We show that if agents have identical preferences, the process is strategyproof. In the general case, we show that an agent who knows the preferences of other agents can find in polynomial time whether he has a strategy for getting a given set of objects, and that if the scoring function is lexicographic, then computing an optimal strategy is polynomial.

2 The general model

2.1 Preferences

We have a set of p indivisible objects $\mathcal{O} = \{o_1, \dots, o_p\}$ and a set of agents $\{1, \dots, n\}$. We assume that each agent i has a (hidden) additive utility function u_i over objects: for any object o, $u_i(o)$ is the value that agent i gives to object o, and for any $A \subseteq \mathcal{O}$, $u_i(A) = \sum_{o \in A} u(o)$ (and $u(\varnothing) = 0$). Therefore, we assume that agents have no preferential dependencies between objects. The preference relation $\succeq_i$ induced by u_i is defined by $A \succeq_i B$ if and only if $u_i(A) \geq u_i(B)$. The strict preference $\succ_i$ associated with $\succeq_i$ is defined as usual by $A \succ B$ if $A \succeq B$ and not $B \succeq A$.

Now, because the allocation process does not involve any elicitation stage, the central authority will never know the agent's utilities. Instead, it can only observe a small part of it, by seeing an agent picking a given objet at a given stage. Now, the central assumption of this paper is that we consider a utility model where any rank in $\{1, \dots, k\}$ is mapped to a utility value via a *scoring function*.

Definition 1 *A* scoring function *is a non-increasing function g from $\{1, \dots, p\}$ to $\mathbb{R}^+$: if $i \leq j$ then $g(i) \geq g(j)$.*

We focus on three prototypical scoring functions:

[1] This sequential process is also considered in [Brams and King, 2005], for characterizing (centralized) efficient allocations in a centralized setting.

- *Borda*: for any k, $g_B(k) = p - k + 1$.
- *lexicographic*: for any k, $g_L(k) = 2^{p-k}$.
- *QI (quasi-indifferent)*: for any k, $g_I(k) = 1 + \varepsilon.(p - k)$, where $\varepsilon \ll 1$.

We denote by $rank_i(o) \in \{1, \dots, p\}$ the rank of object o in the preference relation of agent i. $g(i)$ denotes the value that an agent gives to her i^{th} preferred object: thus, fixing the scoring function to g amounts to assume that $u_i(o) = g(rank_i(o))$, for any agent i and any object o.

The choice of a scoring function depends on the application domain. In a domain where the number of objects received is of primary importance QI is reasonable. On the other extremity, if agents are likely to have huge discrepancies in the way they value objects, the lexicographic model is more realist. Between both, the Borda model (named after the Borda voting rule) assumes that the value agents give to objects decreases linearly between two consecutive ranks. The choice of a scoring function for a specific domain may be guided by some learning process (one may even think of giving different scoring functions to agents according to their type).

A *profile* R consists in a collection of rankings, one for each agent: $R = \langle \succ_1, \dots, \succ_n \rangle$.

2.2 Uncertainty over profiles

First, we assume that for a given agent, all possible rankings are equally probable. Next, we have to specify whether the events "agent i having ranking R_i" and "agent j having ranking R_j" are independent or not. For this we focus on two prototypical models which lie at both extremities of the spectrum: one where these events are independent and one where they are fully correlated.

Full independence (FI) For any agent i, all possible rankings on $\mathcal{O}$ are equiprobable and the rankings of different agents are independent: $Pr(R) = \frac{1}{(p!)^n}$ for every profile $R = \langle \succ_1, \dots, \succ_n \rangle$.

Full correlation (FC) The agents rank the objects in the same way: $\succ_1 = \dots = \succ_n$. This assumption (also considered in [Brams and Fishburn, 2002]) makes sense if the agents are similar enough so that the value of an object can be considered objective. We will see later that this assumption is equivalent to focusing on the worst case, as it gives the worse possible utility to the agents; hence we don't need to define a probability distribution in this case (for the sake of defining the model completely, we may further assume that all profiles of the form $R = \langle \succ, \dots, \succ \rangle$ are equiprobable, hence have probability $\frac{1}{p!}$ each; but the results do not depend on this).

We could have a more general model, with some intermediate correlation between the rankings (*e.g.*, for some constant $\alpha \in [\frac{1}{2}, 1]$, $Pr(o \succ_i o' \mid o \succ_j o') = \alpha$) of which (FI) and (FC) are particular cases; we will not develop it here.

2.3 Policies

At each stage of the process, a designated agent picks an object (supposedly, her preferred object among those that remain), following a *policy* that assigns an agent to each stage. Formally, a *policy* is a function $\pi : [\![1, p]\!] \to \mathcal{N}$. We simply denote a policy by enumerating the agents picking an object at time $1, 2, \dots, p$. For instance, if $n = 3$ and $p = 7$, the policy defined by $\pi(1) = 2, \pi(2) = 1, \pi(3) = 1, \pi(4) = 2, \pi(5) = 3, \pi(6) = 3, \pi(7) = 3$ is denoted by 2112333.

Given a policy π and a profile $R = \langle \succ_1, \dots, \succ_n \rangle$, for every agent i, we denote by $s^{\pi}_{i,k}(R)$ her current share right after stage k. For every i, $s^{\pi}_{i,k}(R)$ is defined inductively as follows:

- $s^{\pi}_{i,0}(R) = \varnothing$
- $s^{\pi}_{i,k}(R) = s^{\pi}_{i,k-1}(R)$ if $\pi(k) \neq i$, and $s^{\pi}_{i,k}(R) = s^{\pi}_{i,k-1}(R) \cup \{\max_{\succ_i}(o \in \overline{\mathcal{O}^{\pi}_{k-1}}(R))\}$ otherwise.
- $S(i, \pi, R) = s^{\pi}_{i,p}(R)$ is the set of items allocated to i at the end of the process,

where $\mathcal{O}^{\pi}_k(R) = \bigcup_i s^{\pi}_{i,k}(R)$ denotes the set of objects already allocated according to π right after stage k, and $\overline{\mathcal{O}^{\pi}_k}(R) = \mathcal{O} \setminus \mathcal{O}^{\pi}_k(R)$ the remaining objects.

Finally, let $u_i(\pi, R) = u_i(S(i, \pi, R))$ be the utility of agent i at the end of the process, that is, the value of her share according to the scoring function:

$$u_i(\pi, R) = \textstyle\sum_{o \in S(i,\pi,R)} g(rank_i(o))$$

Example 1 *Let there be 5 objects, $\pi = 12332$, and 3 agents with the following preferences: $1 : o_1 \succ o_2 \succ o_3 \succ o_4 \succ o_5$; $2 : o_4 \succ o_2 \succ o_5 \succ o_1 \succ o_3$; $3 : o_1 \succ o_3 \succ o_5 \succ o_4 \succ o_2$. The allocation process proceeds as follows:*

k	0	1	2	3	4	5
$s(1)^{\pi}_k$	$\varnothing$	o_1	o_1	o_1	o_1	o_1
$s(2)^{\pi}_k$	$\varnothing$	$\varnothing$	o_4	o_4	o_4	$o_4 o_2$
$s(3)^{\pi}_k$	$\varnothing$	$\varnothing$	$\varnothing$	o_3	$o_3 o_5$	$o_3 o_5$
$\mathcal{O}^{\pi}_k$	$\varnothing$	o_1	$o_1 o_4$	$o_1 o_4 o_3$	$o_1 o_4 o_3 o_5$	$o_1 o_4 o_3 o_5 o_2$

Finally, $S(1, \pi) = \{o_1\}$, $S(2, \pi) = \{o_2, o_4\}$, and $S(3, \pi) = \{o_3, o_5\}$. The utilities of the three agents are the following, depending on the choice made for cardinaliation:

- *Borda:* $u_1(\pi) = 5; u_2(\pi) = 5 + 4 = 9; u_3(\pi) = 4 + 3 = 7.$
- *lexicographic:* $u_1(\pi) = 16; u_2(\pi) = 24; u_3(\pi) = 12.$
- *QI:* $u_1(\pi) = 1 + 4\varepsilon; u_2(\pi) = 2 + 7\varepsilon; u_3(\pi) = 2 + 5\varepsilon.$

2.4 Expected utility and social welfare

Since the arbitrator does not know the agents' preferences, he is not able to compute their actual individual utility, but can only rely on an *expected utility*, given the probability distribution Pr over profiles. Given a policy π, the expected utility of agent i is defined by:

$$\overline{u(i, \pi)} = \sum_{R \in Prof(\mathcal{N}, \mathcal{O})} Pr(R) \times u_i(\pi, R).$$

Finally, we define an *aggregation function* as a symmetric, non-decreasing function from $(\mathbb{R}^+)^n$ to $\mathbb{R}^+$.

Two typical choices for F correspond to the well-known utilitarian criterion and the Rawlsian egalitarian criterion:

- utilitarian: $F(u_1, \dots, u_n) = \sum_{i=1,\dots,n} u_i$;
- egalitarian: $F(u_1, \dots, u_n) = \min_{i=1,\dots,n} u_i$.

Given a probability distribution Pr on profiles and an aggregation function F, the expected social welfare of policy π is defined as the aggregation of individual expected utilities:

$$\overline{sw_F(\pi)} = F(\overline{u(1,\pi)}, \dots, \overline{u(n,\pi)}).$$

Note that $\overline{sw_F(\pi)}$ is determined from the scoring function g, the correlation model c, and the aggregation function F.

To sum up, a *sequential allocation problem* is a 5-uple $P = \langle \mathcal{N}, \mathcal{O}, g, c, F\rangle$ where $\mathcal{N} = \{1, \dots, n\}$ is the set of agents, $\mathcal{O} = \{o_1, \dots, o_p\}$ the set of objects, g the scoring function, $c \in \{FI, FC\}$ the correlation function, and F the aggregation function. A policy π is *optimal* for P if it maximizes $\overline{sw_F(\pi)}$. Solving a sequential allocation problem consists in finding the optimal sequence once those five parameters have been fixed.

3 Computing optimal sequences

3.1 Full correlation

Recall that under the full correlation assumption, all agents have the same ranking over objects. Without of loss of generality, assume this ranking is $o_1 \succ o_2 \succ \dots \succ o_p$. Then, at stage k, the designated agent $\pi(k)$ will pick object o_k. Therefore, expected social welfare can be rewritten as follows.

$$\overline{sw_{F,FC}(\pi)} = F\left(\sum_{k\in\pi^{-1}(1)} g(k), \dots, \sum_{k\in\pi^{-1}(n)} g(k)\right)$$

Note that maximizing this social welfare comes down to maximizing the social welfare for the worst possible profile: let R be a profile with identical preference rankings, and R' be any other profile. Then for any policy π and agent i, $u_i(\pi, R) \leq u_i,(\pi, R')$, hence $\overline{sw_{F,FC}(\pi)} = F(u_1(\pi, R), \dots, u_n(\pi, R)) \leq F(u_1(\pi, R'), \dots, u_n(\pi, R'))$. This is simply because: (i) obviously, at every stage i, agent $\pi(i)$ will get an object she ranks in position at most i, and (ii) under full correlation, agent $\pi(i)$ will actually get the object she ranks i^{th}, which is the worst she could get.

Now we consider the three distinguished scoring functions. We start by utilitarian social welfare. We have

$$\overline{sw_+(\pi)} = \sum_{i\in[\![1,n]\!]} \sum_{k\in\pi^{-1}(i)} g(k) = \sum_{k\in[\![1,p]\!]} g(k)$$

Note that $\sum_{k\in[\![1,p]\!]} g(k)$ is a constant, which depends only n, p, and g, but not on π. In other words:

Proposition 1 *Under utilitarianism and full correlation, all policies have the same expected social welfare.*

Proof Whichever agent is designated by π at stage k, she will pick o_k and receive $g(k)$. Therefore, the total social welfare is $\sum_{k\in[\![1,p]\!]} g(k)$, which does not depend on π. ■

Therefore, under utilitarianism and full correlation, the problem is trivial. Now, we consider egalitarianism and study the following problem:

Problem 1:	Sequential allocation under egalitarianism and full correlation.
INSTANCE :	A number of agents n, a number of objects p, a scoring function g, an integer K.
QUESTION :	Is there a policy π such that $\overline{sw_{\min}(\pi)} \geq K$, under full correlation ?

Proposition 2 *Under egalitarianism and full correlation, Problem 1 is NP-complete.*

The hardness part of the proof, which is not particularly difficult, comes from a reduction from PARTITION.

Proof Since for all π, $\overline{sw_{\min}(\pi)}$ can be computed in polynomial time, the problem is obviously in NP.

NP-hardness is proved by reduction from [PARTITION]: given a set $\mathcal{S}$ and a function $w : \mathcal{S} \to \mathbb{N}$, is it possible to find a subset $\mathcal{S}' \subset \mathcal{S}$ such that $w(\mathcal{S}') = w(\mathcal{S} \setminus \mathcal{S}')$, where $w(\mathcal{X}) = \sum_{x\in\mathcal{X}} w(x)$. Without loss of generality we assume that the integers in S are ordered non-decreasingly, *i.e.*, $S = \{s_1, \ldots, s_p\}$ with $w(s_1) \leq \ldots w(s_p)$. We map every instance $\langle \mathcal{S}, w\rangle$ of PARTITION into the following instance of Problem 1: $\langle 2, |\mathcal{S}|, w, w(\mathcal{S})/2\rangle$, Let $\Pi_{S'}$ mapping every subset $\mathcal{S}' \subseteq \mathcal{S}$ to the following policy: $k \mapsto \{1$ if $k \in \mathcal{S}',\ 2$ otherwise$\}$. Obviously, we have $u_1(\Pi(\mathcal{S}')) = w(\mathcal{S}')$ and $u_2(\Pi(\mathcal{S}')) = w(\mathcal{S}\setminus\mathcal{S}')$. If there is a subset $\mathcal{S}' \subset \mathcal{S}$ such that $w(\mathcal{S}') = w(\mathcal{S} \setminus \mathcal{S}')$, we obviously have $u_1(\Pi(\mathcal{S}')) = u_2(\Pi(\mathcal{S}')) = w(\mathcal{S})/2$. Conversely, every policy π such that $\overline{sw_{\min}(\pi)} \geq \sum_{x\in\mathcal{S}} w(x)/2$ must be such that $u_1(\pi) = u_2(\pi) = \sum_{x\in\mathcal{S}} w(x)/2$. Since $w(\Pi^{-1}(\pi)) = u_1(\pi) = u_2(\pi) = w(\mathcal{S} \setminus \Pi^{-1}(\pi))$, $(\Pi^{-1}(\pi), \mathcal{S} \setminus \Pi^{-1}(\pi))$ is a suitable partition for the initial problem, which proves the validity of the reduction. ■

We now consider the three specific scoring functions defined above.

3.1.1 Lexicographic scoring

It is not difficult to show that if there are at least as many objects as agents ($p \geq n$), the optimal policies are those of the following form, where σ is a permutation of $\{1, \ldots, n\}$: $\sigma(1)\sigma(2)\ldots\sigma(n-1)\sigma(n)^{p-n+1}$ and that if there are less objects than agents ($p < n$), the optimal policies are those of the following form, where σ is a permutation of $\{1, \ldots, n\}$: $\sigma(1)\sigma(2)\ldots\sigma(p)$.

In other terms, the first $n-1$ agents choose an object in sequence, and the remaining agent picks all remaining objects.

3.1.2 Borda scoring

In this case, the problem is equivalent to finding a partition of $\{1, \ldots, p\}$ in n classes such that the sum of the integers in each class is above a threshold. This comes down to solve the following problem:

Problem 2: Sequential allocation under egalitarianism, full correlation, and Borda scoring

INSTANCE : A number of agents n, a number of objects p, an integer K.

QUESTION : Is there a partition of $X_p = [\![1, p]\!]$ into n clusters $A_1, \ldots, A_n$ such that $\sum_{k\in A_i} k \geq K$, $\forall i \leq n$?

Because the number of possible cumulated values for a given agent is polynomially bounded (namely $\frac{p(p+1)}{2}$), the problem can be solved by dynamic programming, where we need to fill a table of $O(p \times n.p^2)$ cells, each requiring $O(n)$ computation time. Thus:

Proposition 3 *Under egalitarianism, full correlation, and Borda scoring, Problem 2 is in P.*

3.1.3 QI scoring

Let $m = \lfloor \frac{p}{n} \rfloor$ and $q = p - nm$. An optimal policy must be such (a) that every agent receives at least m objects and (b) the number of agents receiving m objects is minimal, and (c) the agents who receive only m objects should pick all their objects before those who get more objects. (a) and (b) imply that exactly q agents receive $m+1$ objects, exactly $n-q$ objects receive m objects.

Formally, an optimal policy π is necessarily the form $\pi = \pi_1; \pi_2$ where

- π_1 is a subpolicy of length $(n-q)m$, and π_2 is a subpolicy of length $q(m+1)$;

- there exists a partition of $\mathcal{N}$ into two subsets X and Y, such that $|X| = n - q$ and $|Y| = q$, and such that π_1 involves only agents of X and π_2 involves only agents of Y.
- every agent of X appears exactly m times in π_1 and every agent of Y appears at least $m + 1$ times in π_1.

For instance, if $p = 10$ and $n = 4$ then $m = 2$ and $q = 2$: Examples of policies satisfying these conditions are $\pi = 1122333444$, $\pi' = 1221333444$, and $\pi'' = 1221344343$. Now, not all policies of this form are optimal: π is not optimal, while π' and π'' are optimal. The optimization problem is somehow similar as for the case of Borda. Obviously, what counts is that π_1 is an optimal policy for $n-q$ agents and m items; as long as this is satisfied and π_2 assigns $m + 1$ items to the remaining agents, in any order, this will do — thus, π' is optimal, although the utility gap between agents 2 and 3 is higher than in π''. Therefore, the problem comes down to the problem of computing an optimal policy for p items and n agents where p is a multiple of n. Lastly, when p is a multiple of n, we know that an optimal policy will give the same number of items to everyone, therefore in $1 + k\varepsilon$ we can forget the 1 and we come down to finding an optimal policy for the Borda scoring function. Therefore: *finding an optimal policy for QI comes down to finding an optimal policy for Borda when the number of objects is a multiple of the number of agents.* As a corollary:

Proposition 4 *Under egalitarianism, full correlation, and QI scoring, every policy of the following form is optimal: $\pi = \pi_1; \pi_2$ where $m = \lfloor \frac{p}{n} \rfloor$, $q = p - nm$, and*

1. *π_1 is a subpolicy of length $(n - q)m$, and π_2 is a subpolicy of length $q(m + 1)$;*
2. *there exists a partition of $\mathcal{N}$ into two subsets X and Y, such that $|X| = n-q$ and $|Y| = q$, and such that π_1 involves only agents of X and π_2 involves only agents of Y.*
3. *every agent of X appears exactly m times in π_1 and every agent of Y appears exactly $m+1$ times in π_2.*
4. *π_1 maximizes the Borda scoring (under egalitarianism and full correlation) among all policies giving exactly m objects to each one of $n - q$ agents.*

Proof We first show that all the policies of the above form have the same egalitarian social welfare. For each such policy π, every agent in X gets a utility $m + O(\varepsilon)$ whereas every agent in Y gets a utility $m + 1 + O(\varepsilon)$, therefore, $sw(\pi) = m + O(\varepsilon)$, and the least satisfied agent is in X. For any $i \in X$, $u_i^{QI}(\pi_1) = m+\varepsilon.u_i^{Borda}(\pi_1)$ and $u_i^{QI}(\pi) = m+q(m+1)\varepsilon+\varepsilon.u_i^{Borda}(\pi_1)$. Therefore, if α is the egalitarian social welfare of a policy giving exactly m objects to each one of $n-q$ agents, and maximizing egalitarian social welfare under Borda scoring, then $sw_{\min}^{QI}(\pi) = m + q(m+1)\varepsilon + \varepsilon.\alpha = m + (q(m+1) + \alpha).\varepsilon = s^*$, which is a constant value.

If π satisfies conditions 1, 2 and 3 above but not condition 4, then $sw_{\min}^{QI}(\pi) = m + q(m+1)\varepsilon + \varepsilon.sw^{Borda}(\pi_1) < s^*$. Therefore, (i) any policy π satisfying (1), (2) and (3) is such that $sw^{QI}(\pi) \leq s^*$.

It remains to be shown that if π does not satisfify conditions 1, 2 and 3, that is, if it is not of the form $\pi_1; \pi_2$ as above, then $sw(\pi) \leq s^*$.

If π' gives some agent less than m objects then $sw(\pi') < m$, therefore $sw(\pi') < sw(\pi)$. Therefore, we assume now that π' gives every agent at least m objects.

Let A be the set of agents who get exactly m objects in π', and B the set of agents who get more than m objects. Since $p = q+nm$, $|B| \leq q$ and $|A| \geq n-q$. Assume that $|A| = r > n-q$. First, we claim that at least one agent in B gets at least $m+2$ objects. Assume not; then the agents in B get exactly $m+1$ objects. The total number of objects allocated by π' is $mr+(m+1)(n-r) = nm+(n-r) < nm+q = p$, a contradiction. Therefore, at least one agent in B, say j, gets at least $m+2$ objects. Now, let i be the agent in A who gets his final object first. For instance, if $\pi = 1233424144$, then $A = \{1, 2, 3\}$ and $i = 3$. Consider the following policy θ obtained from π' by (a) giving one more object to i and one less object to j (by removing the last occurrence of j in the sequence), and (b) move i's $m+1$ turns at the end of the sequence. In our example, $\theta = 1242414333$. Then the utility of all agents in A increases, therefore the utility of the least satisfied agent increases, that is, (ii) $sw(\theta) \geq sw(\pi')$.

Now, assume that θ is not of the form $\theta_1; \theta_2$ as above. Then consider the policy σ obtained from θ by moving the turns of the agents in A towards the beginning of the sequence — for instance, if $\theta = 1233424144$ then $\sigma = 123321444$. The utility of the agents in A does not decrease, and the least satisfied agent is an agent in A, therefore (iii) $sw(\sigma) \geq sw(\theta)$, and θ satisfies conditions (1), (2) and (3).

From (ii) and (iii) we get that there exists a policy σ satisfying (1), (2) and (3) such that $sw(\sigma) \geq sw(\pi')$, and from (i), $sw(\sigma) \leq s^*$, therefore $sw(\pi') \leq s^*$. ■

Corollary 1 *Under egalitarianism, full correlation, and QI scoring, finding an optimal policy is in P.*

Proof Finding an optimal policy for Borda scoring and egalitarian social welfare can be solved again by dynamic programming; the only difference with Proposition 3 is that we must restrict the search to sequences that give the same number of objects to everyone. ■

3.2 Full Independence

The full independence (FI) case is more complex. We conjecture that the problem of finding an optimal allocation policy under FI, and for any of our three specific scoring functions, is NP-hard, but we do not have a proof. Moreover, we do not even know if, for a given policy π, $\overline{sw_F(\pi)}$ can be computed in polynomial time (we conjecture it is NP-hard as well).

However, it is possible to compute an optimal policy in reasonable time for small numbers of objects using an exhaustive search algorithm: for each possible complete policy, the algorithm computes the social welfare and compares it to the best one found so far. So as to break symmetries, we only consider policies π where $\pi(k) \leq k$ (the first agent in the sequence can only be 1, the second one can be 1 or 2, and so on).

The expected utility for an agent i can computed by developing a search tree: each node is a partial assignment of the objects, and is expanded into (i) one single branch if i is the next agent to choose (she will pick her top object for sure), and (ii) #(remaining objects) branches otherwise (the current agent can possibly pick each one of the remaining objects with uniform probability). Algorithm 1 slightly improves this procedure by expanding several levels at once, if i does not appear during several successive rounds. At each step, a set X of ranks are chosen, and the problem is transformed into a problem with $p - |X|$ objects, and a scoring function built from g, where all the values from $g(X)$ are removed. Here is an example of how this algorithm computes the utility of agent 1 for the sequence 12221, using the Borda scoring function. We use a compact notation for g, namely $g = 12345$ for $g(1) = 1; \ldots, g(5) = 5$; note that $dom(g)$ is the number of objects remaining to be allocated.

- First call: g is 54321 and the added value (Line 3) is $g(1) = 5$.
- Second call: g is 4321 and since 1 is not served during 3 rounds, there are $\binom{4}{3} = 4$ recursive calls (Line 7): for $X = 123$ (value added : 1), $X = 124$ (value added : 2), $X = 134$ (value added : 3) and $X = 234$ (value added : 4). The global added value is thus $(4 + 3 + 2 + 1)/4 = 2.5$.
- Therefore, the value returned is 7.5.

We provide an implementation of this algorithm that computes optimal policies (and their values) once the user has given n, p, g and F, and can be tested online (`http://92.243.17.107:8080/`).

This implementation computes optimal policies in less than a few minutes until around 10 objects (12 objects when $n = 2$). Table 1 shows some results for small n and p. An intriguing result is that for the values of p and n we tested, strict alternation is an optimal strategy for $F = +$; we do not know whether this is true for every p and n. Also, the optimal strategy returned for $p = 12$, $n = 2$ is not a balanced alternation.

Algorithm 1: $\texttt{EU}(\pi, i, g)$

input : A policy π, an agent i, a scoring function g.
output: Expected utility of i under the FI assumption.

if $(dom(g) = \varnothing) \vee (\forall k, \pi(k) \neq i)$ **then return** 0;
if $\pi(1) = i$ **then**
  return $g(1) + \texttt{EU}(k \mapsto \pi(k+1), i, k \mapsto g(k+1))$;
$\lambda \leftarrow \min\{k' \mid \pi(k') = i\} - 1$;
$u \leftarrow 0$;
for $X \subset [\![1, |dom(g)|]\!]$ *such that* $|X| = \lambda$ **do**
  $u \leftarrow u + \texttt{EU}(k \mapsto \pi(k+\lambda), i, g\circ \uparrow_{dom(g)\setminus X})$;
  /* with $\uparrow_Y : k \mapsto k^{\text{th}}$ element of Y in increasing order */
return $u / \binom{|X|}{k}$;

	Egalitarian		Utilitarian	
p	$n=2$	$n=3$	$n=2$	$n=3$
4	1221	1233	1212	1231
5	11222	12332	12121	12312
6	121221	123321	121212	123123
7	1122122	1232133	1212121	1231231
8	12212112	11332232	12121212	12312312
9	112122212	121332321	121212121	123123123
10	1221121221	1231223133	1212121212	1231231231
12	121212122121		121212121212	

Table 1: Optimal sequences for small n and p, under the FI assumption and Borda scoring function.

Things become much harder when p becomes larger. However, we claim that, in this case, the problem loses much of its interest. Informally, when p increases (while n is fixed) then agents receive more and more objects, and it will be more and more easy to find an optimal sequence.

Proposition 5 *Let $p = kn+q$. Under FI and Borda scoring, any policy π of the form $\sigma_1\sigma_2\ldots\sigma_k\theta$, where $\sigma_1,\ldots,\sigma_k$, are permutations of $\{1,\ldots,n\}$, tends to an optimal allocation when $p \to +\infty$ (n being held constant) both for egalitarian and utilitarian social welfare.*

Proof Assume first that p is a multiple of n. Consider the first n stages. The agent who comes first in σ_1 receives p. The one who comes second receives $\frac{p-1}{p}.p + \frac{1}{p}(p-1) = \frac{p^2-1}{p} \sim_{p\to\infty} p$. The one who comes third gets also $\Theta(p)$, and so on: everyone receives $p + O(p^{-1})$. Now, during the next n stages: the agent who comes first in σ_2 gets her second preferred object if none of the $n-1$ agents have taken it, which happens with probability $1 - \frac{n-1}{p-1}$; therefore, she gets an utility at least $\left(1 - \frac{n-1}{p-1}\right).(p-1) = p - 1 + O\left(p^{-1}\right)$. We check that it also applies to every other agent during the execution of σ_2. During σ_3, we check then that every agent receives $p - 2 + O\left(p^{-1}\right)$, and so on until σ_k, where everyone receives $p - k + 1 + O\left(p^{-1}\right)$. Therefore, the total utility received by any agent is at least $[p + (p-1) + \ldots + (p-k+1)] + O\left(p^{-1}\right)$, *i.e.*, $\frac{p^2}{n} + O(1)$. Now, in the best case, for whatever policy, the maximal utility that an agent can receive is the utility corresponding to his preferred k objects, *i.e.*, $p + \ldots + p - k + 1 = \frac{p^2}{n} + O(1)$. Therefore, $\bar{u}(\sigma)$ tends to the expected utility of an optimal policy when $p \to \infty$.

When p is not a multiple of n, the proof is analogous, noticing that the increment of utility brought by the last object picked by an agent during the last subsequence is small compared to the $\frac{p^2}{n}$ utility already gathered. ■

4 Strategical issues

As most collective decision mechanisms, our sequential allocation problems are generally not strategyproof. This can be seen on the very simple example with any cardinalization function, two agents, four objects, the preferences of 1 being $abcd$ and those of 2 being $bcda$, and $\pi = 1221$. If 1 and 2 play sincerely, *i.e.*, pick their preferred object at each step, then the final allocation is $1 \mapsto ac, 2 \mapsto bd$. However, if 1 knows 2's preferences and plays strategically, then he picks b first, then 2 picks c and d, 1 finally picks a an the allocation is $1 \mapsto ab, 2 \mapsto cd$, which makes him better off.

We now address these two questions: (1) is sequential allocation strategyproof for some restricted domains? (2) when it is not, how hard is it for an agent who knows the preferences of the others to compute an optimal strategy? *For both questions, the choice of the social welfare function F and of a probability distribution over profiles is irrelevant (remember that the manipulating agent knows the others' preferences).*

First we define *picking strategies* and *manipulation problems* formally.

Without loss of generality, let 1 be the manipulator. Let π be a policy. Let $i_1,\ldots,i_r$ be the indices such that $\pi(i_j) = 1$, with $1 \leq i_1 < i_2 < \ldots < i_r \leq p$; these will be called the *picking stages* of agent 1. Let $\langle \succ_2,\ldots,\succ_n\rangle$ be the rankings of the other agents, assumed to known by agent 1. A *strategy* for agent 1 is a function $\sigma : \{1,\ldots,r\}$ to $\mathcal{O}$, specifying which object 1 should take at any stage where it is his turn to pick an object.

Some strategies may fail because some object that 1 intends to take has already been taken. We say that a strategy σ is *well-defined* with respect to π and $\langle \succ_2,\ldots,\succ_n\rangle$ if at any stage i_r, object $o_{\sigma(r)}$ is still available (assuming 1 follows the strategy σ as long as it is possible, and that 2 to n play sincerely). We also say that σ is *well-defined until stage i* if the above conditions hold for i_r as long as $i_r \leq i$. Obviously, σ is well-defined if it is well-defined until the last stage.

A *manipulation problem* M (for agent 1) consists of π, $\langle \succ_2, \ldots, \succ_n \rangle$, and a target set of objects $S \subseteq \mathcal{O}$. A well-defined strategy σ is *successful* for M if, assuming the agents 2 to n act sincerely, σ ensures that agent 1 gets all objects in S. Solving M consists in determining whether there exists a successful strategy.

Note that the allocation process (who gets what and when) and *a fortiori*, the final allocation (who gets which object in the end) is fully determined by the preferences of agents $2, \ldots, n$ (once again, assumed to behave sincerely) and a well-defined picking strategy of agent 1. Let $A(M, \sigma)$ be the assignment process induced by M, namely, the function from $\{1, \ldots, p\}$ to $N \times X$, where $A(M, \sigma)(i) = \langle j, o \rangle$ means that at time i, agent j picks object o.

Example 2 *Let $n = 3$, $X = \{o_1, \ldots, o_6\}$, and consider the manipulation problem M be defined by:*

- $\pi = \langle 1, 2, 3, 1, 2, 3 \rangle$;
- $\succ_2 = o_1 \succ o_2 \succ o_3 \succ o_5 \succ o_4 \succ o_6$;
- $\succ_3 = o_2 \succ o_1 \succ o_6 \succ o_5 \succ o_4 \succ o_3$;

The picking stages of 1 are $i_1 = 1$ and $i_2 = 4$. Consider the strategy σ defined by $\sigma(1) = o_1$; $\sigma(2) = o_3$. The allocation process $A(M, \sigma)$ assigns o_1 to 1, then o_2 to 2, then o_6 to 3, then o_3 to 1, then o_5 to 2, then finally o_4 to 3. σ is well-defined, and $A(M, \sigma)(1) = \langle 1, o_1 \rangle$, $A(M, \sigma)(2) = \langle 2, o_3 \rangle$, $A(M, \sigma)(3) = \langle 3, o_6 \rangle$, $A(M, \sigma)(4) = \langle 1, o_3 \rangle$, $A(M, \sigma)(5) = \langle 2, o_5 \rangle$, and $A(M, \sigma)(6) = \langle 3, o_4 \rangle$.

Consider now σ' defined by $\sigma(1) = o_1$; $\sigma(2) = o_2$. The allocation process starts by assigning o_1 to 1, then o_2 to 2, then o_6 to 3, then tries to assign o_2 to 1, which is nt possible since o_2 is no longer available. Therefore, σ is not well-defined, and $A(M, \sigma)$ is undefined.

Now, back to Question (1): we show that the answer is positive when the agents' rankings coincide.

Proposition 6 *Under the full correlation assumption, sequential allocation is strategyproof for any scoring function (and for any sequence π).*

Proof Assume without loss of generality that the preference relation of every agent is $o_1 \succ \ldots \succ o_p$. Let $i_1, \ldots, i_m$ the indices in the sequence π such that $\pi(i_1) = \ldots = \pi(i_m) = 1$, and $i_1 < \ldots < i_m$. If all agents are sincere, then 1 gets $\{o_{i_1}, \ldots, o_{i_m}\}$. Assume now that 1 does not play a sincere strategy, while the others keep playing sincere. Let i_k be the first stage where 1 does not play sincerely, and o_j be the object picked by 1 at stage i_k. Obviously, $j > i_k$. Let B be the set of objects that 1 gets using his strategy and $A = \{o_{i_1}, \ldots, o_{i_m}\}$ the set of objects he would get using the sincere strategy. We want to show that $u_1(B) < u_1(A)$. We do this by backward induction on k. Formally, the induction hypothesis H_k is: if the first stage where 1 plays not sincerely is i_k, then whatever he plays next, and assuming that all other agents play sincerely, he will get a lower utility than if he had played sincerely.

The base case is $k = m$. We have $A = \{o_{i_1}, \ldots, o_{i_{m-1}}, o_{i_m}\}$ and $B = \{o_{i_1}, \ldots, o_{i_{m-1}}, o_j\}$. Because $j > i_m$, for any scoring function we have $u_1(o_j) < u_1(o_{i_m})$, therefore $u_1(B) < u_1(A)$, which shows that H_m is true.

Now, let $k < m$, assume $H_{k+1}, \ldots, H_m$ are true and let us show that H_k is true. We consider two cases, depending whether the object o_j picked by 1 at stage i_k is preferred to the next object that 1 would get in the sincere strategy or not.

Case 1: $j < i_{k+1}$ Because the other agents are sincere, at stage $i_k + 1, i_k + 2, \ldots, i_{k+1} - 1$ they will pick $o_{i_k}, o_{i_k+1}, \ldots, o_{j-1}, o_{j+1}, \ldots, o_{i_{k+1}} - 1$, and when it is again 1's turn to pick an object, the set of remaining objects is $C = \{o_{i_{k+1}}, \ldots, o_p\}$. Because C is also the set of objects that 1 has to choose from if he had played sincerely at stage i_k, using the induction hypothesis we know that $u_1(A \cap C) \geq u_1(B \cap C)$ (which is actually a strict inequality if 1 plays not sincere at least one more time, and an equality otherwise). Since $u_1(o_{i_k}) > u_1(o_j)$, we get $u(A) > u(B)$.

Case 2: $j \geq i_{k+1}$ Let q such that $(i_{k+1} \leq) i_q \leq j < i_{q+1}$. Since 1 does not take object o_{i_k}, another agent will take it, the next one will take $o_{i_k} + 1$ and so on, until it is again 1's turn to pick an object. But from then on, the induction hypothesis applies and 1 has to play sincerely to maximize his utility; therefore he will pick $o_{i_{k+1}-1}$, $o_{i_{k+2}-1}$, ..., $oi_q - 1$, and then $o_{i_{q+1}}$, ..., o_{i_m}. So, from step i_k on agent 1 finally gets $S_1 = \{o_j, o_{i_{k+1}-1}, o_{i_{k+2}-1}, \ldots, oi_q - 1, o_{i_{q+1}}, \ldots, o_{i_m}\}$ instead of $S_2 = \{o_{i_k}, o_{i_{k+1}}, o_{i_{k+2}}, \ldots, oi_q, o_{i_{q+1}}, \ldots, o_{i_m}\}$ that he would have got if he had followed his sincere strategy. Now, we can compare the objects in S_1 and S_2 pairwise as follows: $o_{i_k} \succeq_1 o_{i_{k+1}-1}$; $o_{i_{k+1}} \succeq o_{i_{k+2}-1}$; ...; $o_{i_{q-1}} \succeq o_{i_q-1}$; $o_{i_q} \succ o_j$; $o_{i_{q+1}} \sim o_{i_{q+1}}$; etc. (the remaining objects are the same in both lists). Therefore $S_2 \succ_1 S_1$: thus agent 1 gets a lower utility if he does not plays a sincere strategy at step i_k, which shows that H_k is true.

We now move to the Question 2. Below we show that the manipulation problem can be solved in polynomial time. First, we give a simple characterization of successful strategies in problems with two agents.

Proposition 7 *Let μ be the permutation of $\{1, \ldots, p\}$ such that $\succ_2 = o_{\mu(1)} \succ \ldots \succ o_{\mu(p)}$. For any $j \leq p$, let $PS(j) = \#\{i \leq j | \pi(i) = 1\}$ be the number of picking stages of 1 until j and $Cl(j) = \{o_{\mu(i)} | i \leq j, o_{\mu(i)} \in S\}$. There exists a successful strategy for 1 iff for any $j \leq p$ we have $PS(j) \geq |Cl(j)|$. Moreover, in this case any strategy starting by picking the objects in S according to their ranking in $\succ_2$ (and completed so as to be well-defined) is successful.*

Proof Let $\succ_2 = o_{\mu(1)} \succ \ldots \succ o_{\mu(p)}$. For all j, let $C(j) = |Cl(j)|$. Intuitively, $Cl(j)$ represents the objects in S "claimed" by agent 2 until j.

Assume $PS(j) \geq C(j)$ for any $j \leq p$. This implies that the number of picking stages of 1 is at least $|S|$. Consider a strategy σ that starts by picking the objects in S, and according to their ranking in $\succ_2$: $\sigma(1)$ is the first object in S appearing in $\succ_2$, $\sigma(2)$ is the second one, etc. Let $j_1, \ldots, j_{|S|}$ be the $|S|$ picking stages of 1. Let σ be the picking strategy defined as above.

We prove by induction on j that just before stage j, 2 has picked only objects in $\overline{S}$. This is obviously true for $j = 1$. Suppose it is true for $j < j_{|S|}$ and consider stage $j + 1$. If $\pi(j + 1) = 1$ then the induction hypothesis is obviously true for $j + 1$. Let $\pi(j + 1) = 2$. Let $r = PS(j)$ be the number of picking stage for 1 until j included. The number of picking stages for 2 until j included is $j - r$, therefore $j + 1$ is the $(j-r+1)^{\text{th}}$ picking stage for 2. Because so far 2 has picked only objects in $\overline{S}$, and because 1 has picked only objects in S (because of the definition of σ and the fact that $j < j_{|S|}$), we know that at stage j, the objects that have been taken are the first r objects in S (taken by 1) and the first $j - r$ objects in $\overline{S}$, and at stage $j+1$ agent 2 will pick the preferred object after these ones. We claim that this object is not in S. Indeed, if it were, then there would be $j - (j - r) + 1 = r + 1$ objects in S among the first $j + 1$, and since $\pi(j + 1) = 2$, $PS(j + 1) = r$, therefore $C(j + 1) > r$, which contradicts the assumption $PS(j) \geq C(j)$. Thererore, the next object 2 will pick is in $\overline{S}$, and again the induction hypothesis is true for $j + 1$. Finally, applying the induction hypothesis to $j_{|S|}$, we get that σ is successful. Now, assume $C(j) > PS(j)$ for some $j \leq p$, and let j^* be the smallest such integer. We claim that $C(j^*) = C(j^* - 1) + 1 = PS(j^*) + 1 = PS(j^* - 1) + 1*$. First, $PS(j)$ and $C(j)$ are incremented at most by 1 at each stage, *i.e.*, $PS(j) \leq PS(j + 1) \leq PS(j) + 1$ and $C(j) \leq C(j + 1) \leq C(j) + 1$. Consider the four possible cases: (a) $C(j^*) = C(j^* - 1) + 1$ and $PS(j^*) = PS(j^* - 1) + 1$; (b) $C(j^*) = C(j^* - 1)$ and $PS(j^*) = PS(j^* - 1) + 1$; (c) $C(j^*) = C(j^* - 1)$ and $PS(j^*) = PS(j^* - 1)$; (d) $C(j^*) = C(j^* - 1) + 1$ and $PS(j^*) = PS(j^* - 1)$. In cases (a), (b) and (c) we would have $C(j^* - 1) > PS(j^* - 1)$, therefore j^* would not be the smallest integer for which the property $C(j) > PS(j)$ holds. Therefore only (d) is possible. Now, if we had $C(j^*) \geq PS(j^*) + 2$ then we would have $C(j^* - 1) = C(j^*) - 1 \geq PS(j^*) + 1 = PS(j^* - 1) + 1$, therefore, again we would have $C(j^* - 1) > PS(j^* - 1)$. Therefore, $C(j^*) = PS(j^*) + 1$. Putting everything together, we have $PS(j^*) = PS(j^* - 1)$, $C(j^*) = PS(j^*) + 1$ and $C(j^* - 1) = PS(j^* - 1) = PS(j^*) = C(j^*) - 1$.

This implies that among the preferred $j^* - 1$ objects by 2, there are $PS(j^* - 1)$ in S and $j^* - 1 - PS(j^* - 1)$ in $\overline{S}$. Now, because $C(j^*) = PS(j^*) + 1$, the next object picked by 2 will be in S, therefore σ is not successful, and by Lemma 1, no other strategy is successful. ■

Example 3 *Let :*

- $n = 3$;
- $X = \{o_1, \ldots, o_{12}\}$;
- $\pi = \langle 1, 2, 2, 1, 2, 2, 1, 2, 2, 1, 2, 2 \rangle$;

- $\succ_2 = o_7 \succ o_8 \succ o_1 \succ o_{10} \succ o_9 \succ o_{12} \succ o_2 \succ o_6 \succ o_3 \succ o_{11} \succ o_5 \succ o_4$;
- $S = \{1, 2, 3, 4\}$.

stage j	*first* j *objects in* $\succ_*$	$PS(j)$	$Cl(j)$	$PS(j) \leq \|Cl(j)\| = C(j)$?
1	o_7	1	$\varnothing$	*yes*
2	$o_7 o_8$	1	$\varnothing$	*yes*
3	$o_7 o_8 o_1$	1	o_1	*yes*
4	$o_7 o_8 o_1 o_{10}$	2	o_1	*yes*
5	$o_7 o_8 o_1 o_{10} o_9$	2	o_1	*yes*
6	$o_7 o_8 o_1 o_{10} o_9 o_{12}$	2	o_1	*yes*
7	$o_7 o_8 o_1 o_{10} o_9 o_{12} o_6$	3	o_1	*yes*
8	$o_7 o_8 o_1 o_{10} o_9 o_{12} o_6 o_2$	3	$o_1 o_2$	*yes*
9	$o_7 o_8 o_1 o_{10} o_9 o_{12} o_6 o_2 o_3$	3	$o_1 o_2 o_3$	*yes*
10	$o_7 o_8 o_1 o_{10} o_9 o_{12} o_6 o_2 o_3 o_{11}$	4	$o_1 o_2 o_3$	*yes*
11	$o_7 o_8 o_1 o_{10} o_9 o_{12} o_6 o_2 o_3 o_{11} o_5$	4	$o_1 o_2 o_3$	*yes*
12	$o_7 o_8 o_1 o_{10} o_9 o_{12} o_6 o_2 o_3 o_{11} o_5 o_4$	4	$o_1 o_2 o_3 o_4$	*yes*

Therefore, a successful picking strategy for 1 is $\sigma(1) = o_1; \sigma(2) = o_2; \sigma(3) = o_3; \sigma(4) = o_4$.

Consider now manipulation problem obtained from M *by changing* S *into* $S = \{o_8, o_9, o_{10}\}$. *Now we get*

stage j	*first* j *objects in* $\succ_*$	$PS(j)$	$Cl(j)$	$PS(j) \leq \|Cl(j)\| = C(j)$?
1	o_7	1	$\varnothing$	*yes*
2	$o_7 o_8$	1	o_8	*yes*
3	$o_7 o_8 o_1$	1	o_8	*yes*
4	$o_7 o_8 o_1 o_{10}$	2	$o_8 o_{10}$	*yes*
5	$o_7 o_8 o_1 o_{10} o_9$	2	$o_1 o_{10} o_9$	*no*

We can stop here: there is no successful strategy.

Now, we show that a problem M with n agents can be translated into a problem M^* with two agents, such that there is a successful strategy in M if and only if there is a successful strategy in M^*. M^* is defined as follows:

- $S^* = S$;
- the preference relation $\succ_*$ of agent 2 is computed by Algorithm 2,
- the policy π^* is defined by: for every $i \leq p$, if $\pi(i) = 1$ then $\pi^*(i) = 1$, and if $\pi(i) > 1$ then $\pi^*(i) = 2$.

Example 4 *We first show how this algorithm works on an example.*

- $n = 3$;
- $X = \{o_1, \ldots, o_{12}\}$;
- $\pi = \langle 1, 2, 3, 1, 2, 3, 1, 2, 3, 1, 2, 3 \rangle$;
- $\succ_2 = o_7 \succ o_8 \succ o_1 \succ o_{10} \succ o_{12} \succ o_3 \succ o_2 \succ o_{11} \succ o_5 \succ o_4 \succ o_9 \succ o_6$;
- $\succ_3 = o_8 \succ o_9 \succ o_2 \succ o_1 \succ o_6 \succ o_5 \succ o_4 \succ o_{12} \succ o_{11} \succ o_{10} \succ o_7 \succ o_3$;

Algorithm 2: Transforms a n-agent manipulation problem into a 2-agent manipulation problem

input : $\langle \succ_2, \ldots, \succ_n \rangle$: preference rankings; $\mathcal{O}$: set of objects; $S \subseteq \mathcal{O}$: target subset; π: policy
output: a preference relation $\succ_*$ on X

$T \leftarrow \mathcal{O}$; $i \leftarrow 1$; $\succ^* \leftarrow \varnothing$; `/* Initialization */`
repeat
  $j \leftarrow \pi(i)$; `/* agent` j `is the next one to pick an object */`
  if $j \neq 1$ **then** `/* this agent is not the manipulator */`
    $o_l \leftarrow Max(\succ_j, T)$; `/*` j `intends to pick` o_l `*/`
    append o_l to $\succ_*$;
    $T \leftarrow T \setminus \{o_l\}$;
    if $o_l \notin S$ **then**
      $i \leftarrow i + 1$; `/* next agent in the sequence */`
      `/* only if 1 and` j `do not compete on object` o_l `*/`
  else $i \leftarrow i + 1$;
  if $i = p + 1$ **then**
    complete $\succ_*$ with all T, in arbitrary order;
    $T \leftarrow \varnothing$;
until $T = \varnothing$;

- $S = \{1, 2, 3, 4\}$.

We get, step after step:

- *initialization:* $T = \{o_1, \ldots, o_{12}\}$, $i = 1$, $\succ_* = \varnothing$;
- *step 1:* $j \leftarrow \pi(1) = 1$; $i \leftarrow 2$;
- *step 2:* $i = 2$; $j \leftarrow \pi(2) = 2$; $o_l = o_7$; $\succ_* = [o_7]$; $T \leftarrow \{o_1, \ldots, o_6, o_8, \ldots, o_{12}\}$; $i \leftarrow 3$;
- *step 3:* $i = 3$; $j \leftarrow \pi(3) = 3$; $o_l = o_8$; $\succ_* = [o_7 \succ o_8]$; $T \leftarrow \{o_1, \ldots, o_6, o_9, \ldots, o_{12}\}$; $i \leftarrow 4$;
- *step 4:* $i = 4$; $j \leftarrow \pi(4) = 1$; $i \leftarrow 5$;
- *step 5:* $i = 5$; $j \leftarrow \pi(5) = 2$; $o_l = o_1$; $\succ_* = [o_7 \succ o_8 \succ o_1]$; $T \leftarrow \{o_2, \ldots, o_6, o_9, \ldots, o_{12}\}$; $i \leftarrow 4$; *i remains unchanged;*
- *step 6:* $i = 5$; $j = 2$; $o_l = o_{10}$; $\succ_* = [o_7 \succ o_8 \succ o_1 \succ o_{10}]$; $T \leftarrow \{o_2, \ldots, o_6, o_9, o_{11}, o_{12}\}$; $i \leftarrow 6$;
- *step 7:* $i = 6$; $j \leftarrow 3$; $o_l = o_9$; $\succ_* = [o_7 \succ o_8 \succ o_1 \succ o_{10} \succ o_9]$; $T \leftarrow \{o_2, \ldots, o_6, o_{11}, o_{12}\}$; $i \leftarrow 6$;
- *step 8:* $i = 7$; $i \leftarrow 8$;
- *step 9:* $i = 8$; $j \leftarrow 2$; $o_l = o_{12}$; $\succ_* = [o_7 \succ o_8 \succ o_1 \succ o_{10} \succ o_9 \succ o_{12}]$; $T \leftarrow \{o_2, \ldots, o_6, o_{11}\}$; $i \leftarrow 9$;
- *step 10:* $i = 9$; $j \leftarrow 3$; $o_l = o_2$; $\succ_* = [o_7 \succ o_8 \succ o_1 \succ o_{10} \succ o_9 \succ o_{12} \succ o_2]$; $T \leftarrow \{o_3, \ldots, o_6, o_{11}\}$; *$i$ remains unchanged;*

- *step 11: $i = 9$; $j = 3$; $o_l = o_6$; $\succ_* = [o_7 \succ o_8 \succ o_1 \succ o_{10} \succ o_9 \succ o_{12} \succ o_2 \succ o_6]$; $T \leftarrow \{o_3, o_4, o_5, o_{11}\}$; $i \leftarrow 10$;*

- *step 12: $i = 10$; $j = 1$; $i \leftarrow 11$;*

- *step 13: $i = 11$; $j = 2$; $o_l = o_3$; $\succ_* = [o_7 \succ o_8 \succ o_1 \succ o_{10} \succ o_9 \succ o_{12} \succ o_2 \succ o_6 \succ o_3]$; $T \leftarrow \{o_4, o_5, o_{11}\}$; i remains unchanged;*

- *step 14: $i = 11$; $j = 2$; $o_l = o_{11}$; $\succ_* = [o_7 \succ o_8 \succ o_1 \succ o_{10} \succ o_9 \succ o_{12} \succ o_2 \succ o_6 \succ o_3 \succ o_{11}]$; $T \leftarrow \{o_4, o_5\}$; $i \leftarrow 12$;*

- *step 15: $i = 12$; $j = 3$; $o_l = o_5$; $\succ_* = [o_7 \succ o_8 \succ o_1 \succ o_{10} \succ o_9 \succ o_{12} \succ o_2 \succ o_6 \succ o_3 \succ o_{11} \succ o_5]$; $T \leftarrow \{o_4\}$; $i \leftarrow 13$;*

- *step 16: $i = 13$; $\succ_* = [o_7 \succ o_8 \succ o_1 \succ o_{10} \succ o_9 \succ o_{12} \succ o_2 \succ o_6 \succ o_3 \succ o_{11} \succ o_5 \succ o_4]$. Stop.*

Finally, π^* is the strategy for agent 1 in P^* defined by: for every $i \leq p$, if $\pi(i) = 1$ then $\pi^*(i) = 1$, and if $\pi(i) > 1$ then $\pi^*(i) = 2$.

Proposition 8 *There exists a successful strategy for 1 in M if and only if there exists a successful strategy for 1 in M^*.*

The proof of Proposition 8 is structured in two lemmas.

Lemma 1 *If there exists a successful strategy for M then there exists a successful strategy for M in which the first $|S|$ objects picked by 1 are the objects of S.*

This lemma means that it is never harmful for 1 to start picking the objects he wants; picking an object out of S instead will never help.

Proof The proof is constructive. Assume there exists a successful strategy σ for M. Then we build the strategy σ' as follows: $\sigma'(1)$ is the object in S that appears first in σ, $\sigma'(2)$ is the object in S that appears next in σ, and so on, that is, for every $i = 1, \ldots, |S|$, $\sigma(i)$ is the i-th object in S according to the order of appearance in σ. For instance, if $S = \{o_1, o_2, o_3\}$ and $\sigma = o_7 o_5 o_2 o_4 o_1 o_3 o_6$, then $\sigma'(1) = o_2$, $\sigma'(2) = o_3$ and $\sigma'(3) = o_1$. After that, σ' is completed arbitrarily until stage $m + 1$ by objects that are still available. This last step guarantees that if σ' is well-defined until stage $i_{|S|}$ then it is well-defined (until the last stage). We claim that σ' is successful.

We can safely ignore steps prior to i_1, since the objects picked by the agents until then will be the same whether 1 follows σ or σ'. Therefore, without loss of generality, we assume $i_1 = 1$. At every stage $i \geq i_1 = 1$, let o_i (resp. o'_i) the object that agent $\pi(i)$ gets when 1 applies σ (resp. σ'), and let $O_i = \{o_1, \ldots, o_i\}$ and $O'_i = \{o'_1, \ldots, o'_i\}$. We show that for every $i = 1, \ldots, i_{|S|}$, (1) either $\pi(i) = 1$ or $o'_i \succeq_{\pi(i)} o_i$, and (2) $O_i \setminus S \supseteq O'_i \setminus S$. We show this by induction on i. This is true for $i = 1$: since $\pi(1) = 1$, then (1) is trivially true, and $O_1 \setminus S = \{\sigma(1)\} \supseteq O'_1 \setminus S = \varnothing$, because $\sigma'(1) \in S$ by construction of σ'.

Now, assume that the induction hypothesis is true from stage 1 to stage i, with $i \leq i_{|S|} - 1$. Let $\pi(i+1) = j$. If $j = 1$, then $i + 1 = i_r$ for some $r \leq |S|$. (1) is trivially true, and $O_{i+1} \setminus S = (O_i \setminus S) \cup (\{o_{i+1}\} \setminus S)$, $O'_{i+1} \setminus S = (O'_i \setminus S) \cup (\{o'_{i+1}\} \setminus S) = O'_i \setminus S$ (because $o_{i+1} \in S$ by construction of σ'), therefore $O'_{i+1} \setminus S = O'_i \setminus S \subseteq O_i \setminus S \subseteq O_{i+1} \setminus S$. If $j \neq 1$, then assume $o'_{i+1} \prec_j o_{i+1}$. This implies that at stage $i+1$, o_{i+1} is no more available when 1 follows σ', otherwise j would take it instead of o'_{i+1}. Therefore, $o_{i+1} \in O'_i$. Now, recall that σ is successful, hence $o_{i+1} \notin S$ and $o_{i+1} \in O'_i \setminus S$, which by the induction hypothesis $O'_i \setminus S \subseteq O_i \setminus S$, implies $o_{i+1} \in O_i \setminus S$ and *a fortiori*, $o_{i+1} \in O_i$, which is impossible since $O_i = \{o_1, \ldots, o_i\}$, and all o_k's are disjoint (an object cannot be picked twice). Therefore, we have $o'_{i+1} \succeq_j o_{i+1}$.

Now, we show that we cannot have $o'_{i+1} \in S$. Assume $o'_{i+1} \in S$. Let r be the number of picking stages of 1 until stage $i+1$, that is, r is the largest integer such that $i_r < i+1$. Since o'_{i+1} is still available after following σ', it has not been picked by 1 before, which means that $o'_{i+1} \notin \{\sigma'(1), \ldots, \sigma'(r)\}$. Now, by construction of σ' from σ, we have that $\sigma'(1), \ldots, \sigma'(r)$ are all in S (because $i \leq i_{|S|} - 1$), and $\sigma'(k)$ is the object in S that appears kth in σ. Therefore $\{\sigma(1), \ldots, \sigma(r)\} \cap S \subseteq \{\sigma'(1), \ldots, \sigma'(r)\} \cap S = \{\sigma'(1), \ldots, \sigma'(r)\}$. Now, since we assumed that $o'_{i+1} \in S$, if we had $o'_{i+1} \in \{\sigma(1), \ldots, \sigma(r)\}$ then we would have $o'_{i+1} \in \{\sigma(1), \ldots, \sigma(r)\} \cap S$, therefore $o'_{i+1} \in \{\sigma'(1), \ldots, \sigma'(r)\}$, contradiction. Therefore,

$o'_{i+1} \notin \{\sigma(1), \ldots, \sigma(r)\}$. Because σ is successful, we have $o_{i+1} \notin S$. Now, $o'_{i+1} \in S$ and $o_{i+1} \notin S$ imply $o'_{i+1} \neq o_{i+1}$, which together with $o'_{i+1} \succeq_j o_{i+1}$ implies $o'_{i+1} \succ_j o_{i+1}$. $O'_i \setminus S \subseteq O_i \setminus S$ and $|O_i| = |O'_i|$ imply $O'_i \cap S \supseteq O_i \cap S$. Therefore, $o'_{i+1} \notin O_i$. But then, since $o'_{i+1} \neq o_{i+1}$, agent j would have picked o'_{i+1} instead of o_{i+1} when agent 1 follows σ, which is not possible because σ is successful. Therefore, we have shown that $o'_{i+1} \notin S$.

Given this, $O_{i+1} \setminus S \supseteq O'_{i+1} \setminus S$ is equivalent to $o'_{i+1} \in O_i \cup \{o_{i+1}\}$. If we had $o'_{i+1} \notin O_i \cup \{o_{i+1}\}$, that is, $o'_{i+1} \notin O_i$ and $o'_{i+1} \neq \{o_{i+1}\}$, then since $o'_{i+1} \succeq_j o_{i+1}$, agent j would have picked o'_{i+1} instead of o_{i+1} in the allocation process following σ, a contradiction. Therefore $o'_{i+1} \in O_i \cup \{o_{i+1}\}$, which implies that $O_{i+1} \setminus S \supseteq O'_{i+1} \setminus S$. Thus we have established that (1) and (2) hold for stage $i+1$.

Finally, assume σ' is not successful. Then there is a stage $i \leq i_{|S|}$ such that $\pi(i) = 1$, that is, $i = i_r$ for some r, and $\sigma'(r) \in O'_i$. Because $\sigma'(r) \in S$ by construction of σ', we have $\sigma'(r) \in O'_i \cap S$. Now, we know that (2) holds for all $i \leq i_{|S|} - 1$. (2), together with $|O_i| = |O'_i|$, imply $O'_i \cap S \subseteq O_i \cap S$. Therefore, $\sigma'(r) \in O_i \cap S$: in words, when the r-th picking stage of agent 1 comes, $\sigma'(r)$ is no more available. But now, by construction of σ', we have $\sigma'(r) = \sigma(r')$ for some $r' \geq r$. Let i' such that $i_{r'} = i'$. $O_{i'} \supseteq O_i$, because $i' \geq i$. Therefore, $\sigma'(r) \in O_i$ implies $\sigma'(r) \in O_{i'}$: agent 1 cannot pick $\sigma'(r)$ at his r'th picking stage, because $\sigma'(r)$ being in $O_{i'}$, it has already been picked. But this implies σ is not successful, a contradiction. Therefore, σ' is successful.

The second lemma establishes the equivalence (with respect of the existence of successful strategies) between the n-agent problem M and the 2-agent problem M^*.

Lemma 2 *Let θ be a strategy for agent 1 (either for M or M^*) in which the first $|S|$ objects picked by 1 are the objects of S. Then θ is S-successful for M if and only if it is successful for M^*, and in that case, the allocation process induced by M and θ is identical to the allocation process induced by M^* and θ (that is, they assign the same objects to the same agents in the same order).*

Proof Let $M = \langle \pi, \succ_2, \ldots, \succ_n \rangle$ and the associated $M^* = \langle \pi^*, \succ_* \rangle$. Let θ be a successful strategy for agent 1 in M. Without loss of generality (due to Lemma 1), we assume that in θ, the first $|S|$ objects picked by 1 are the objects of S, that is, in which $\{\theta(1), \ldots, \theta(|S|)\} = S$. Let also be $O(M, \theta, i) = \{o | A(M, \theta)(j) = o \text{ for some } j \leq i\}$ be the set of objects already picked at stage i in the allocation process $A(M, \theta)$, and define $O(M^*, \theta, i)$ similarly. Say that an assignment function θ is *successful until* i in M (resp. in M^*) if for any $i' \leq i$, $A(M, i') = (j, o)$ (resp.$A(M^*, i') = (j, o)$) and $o \in S$ imply $j = 1$: if an object in S has already been picked, then it has been picked by agent 1.

We start by proving the following invariant: for every i such that (a) θ is successful until i in M and (b) $i \leq i_{|S|}$, the following three conditions hold:

1. if $\pi(i) = 1$, then $A(M, \theta)(i) = A(M^*, \theta)(i)$;
2. if $\pi(i) = j \neq 1$, then $A(M, \theta)(i) = (j, o)$ if and only if $A(M^*, \theta)(i) = (*, o)$, where $*$ is the artificial agent gathering agents 2 to n.
3. $O(M, \theta, i) = O(M^*, \theta, i)$.

In words: as long as 1 has not picked all his objects, and as none of his objects has been taken by another agent, the allocation processes are equivalent, in the sense that the objects are assigned in the same order, and that objects assigned to 1 are the same. Note that 3 is a consequence of (1) and (2), therefore it is enough to prove only (1) and (2) for every i satisfying (a) and (b).

We prove this by induction on i.

If $i = 1$, then either $\pi(1) = 1$, in which case $A(M, \theta)(1) = A(M^*, \theta)(1) = (1, \theta(1))$ — 1 picks his first object in S, as specified by θ — or $\pi(1) = j \neq 1$; in this case, the preferred object o by j cannot be in S (otherwise θ would not be successful in M until 1); then $A(M, \theta)(1) = (j, o)$, where o is the most preferred object of agent j (j picks his most preferred object); but, by construction of $\succ_*$, o is also the most preferred object of agent $*$, therefore $A(M^*, \theta)(1) = (*, o)$.

Assume conditions (1) and (2) (and therefore (3)) are satisfied until i, and assume that $i+1$ satisfies (a) and (b). If $\pi(i+1) = 1$, there exists a r such that $i + 1 = i_r$. Because θ is successful until $i+1$, $\theta(r)$ must still be available at that stage, that is, there is no $j \neq i$ such that $A(M, \theta)(j) = \theta(r)$; and then $A(M, \theta)(i+1) = \theta(r)$ (at stage $i+1$, agent 1 takes the object $\theta(r)$ as specified in θ). Now, because $O(M, \theta, i) = O(M^*, \theta, i)$, $\theta(r)$ is still available also at stage $i+1$ in the process (M^*, θ), and therefore $A(M^*, \theta)(i+1) = \theta(r) = A(M, \theta)(i+1)$. Now, if $\pi(i+1) = j \neq 1$, then the available object o preferred by j cannot be in S, otherwise θ would not be successful in M until $i+1$. Therefore, $A(M, \theta)(i+1) = A(M^*, \theta)(i+1) = (*, o)$. In both cases, conditions 1 and 2 are satisfied at stage $i+1$, therefore the proof by induction is completed.

Now, assume θ is successful. Then condition (a) is always satisfied and the induction hypothesis is true until $i_{|S|}$, which, applying (1) to $i_{|S|}$, implies that θ is successful in M^*.

Conversely, assume θ is *not* successful in M. We claim that it is not successful in M^* either. Since θ is not successful in M, there must be a stage i such that $\pi(i) = j \neq 1$, and $A(M, \theta)(i) = o \in S$. Let i_{min} be the smallest such integer. We can apply the induction hypothesis up to $i_{min} - 1$, from which we get that $O(P, \theta, i_{min} - 1) = O(M^*, \theta, i_{min} - 1)$. This, because $A(M, \theta)(i) = o$, implies that $o \notin O(M, \theta, i_{min} - 1)$, therefore $o \notin O(M^*, \theta, i_{min})$, and now, at stage i_{min}, by construction of $\succ_*$, agent $*$ picks o, and θ fails in M^*. ■

The proof of Proposition 8 follows easily from Lemmas 1 and 2.

So far we have addressed an important issue: given a target set of objects S and assuming that (a) all other agents plays sincere and that (b) the manipulating agent knows their preferences, then this agent can find, in polynomial time, a strategy for getting all objects in S whenever there exists one. We now go further and address the following issue: when given an additive utility function for agent 1, can agent 1 find an *optimal* strategy in polynomial time?

We show that this is true under the lexicographic cardinalization function. The intuition of the proof goes as follows. Let 1 be the manipulator. We build the best set of objects that 1 can manage to get (assuming the other agents behave sincerely), in a greedy way, by considering the objects one after the other in decreasing order of 1's preference ranking; if we find out that 1 has a strategy to manage to get this object in addition to the already secured objects, then we add this object to the best set of objects that 1 can get; otherwise, we don't, and move on to the following object. This greedy algorithm calls the previous algorithm to check whether there exists a S-successful strategy.

Algorithm 3: Finding the optimal strategy for agent 1

---

input : a policy π; a collection of preference rankings $\langle \succ_1, \succ_2, \ldots, \succ_n \rangle$
output: an optimal picking strategy σ for agent 1

1 $t \leftarrow$ number of occurrences of 1 in π;
2 $S \leftarrow \varnothing$;
3 $i \leftarrow 1$;
4 **repeat**
5 | **if** $\exists \sigma$, *successful strategy for* $S \cup \{i\}$, π *and* $\langle \succ_1, \succ_2, \ldots, \succ_n \rangle$ **then** $S \leftarrow S \cup \{i\}$;
  | $i \leftarrow i + 1$;
6 **until** $i > p$ **or** $|S| = t$;
7 **return** σ

The soundness and completeness of Algorithm 1 is expressed by the following result:

Proposition 9 *If agent 1's utility function is lexicographic, then Algorithm 2 returns the optimal strategy for agent 1.*

Proof Suppose not: there exists a strictly better strategy σ'. Let k be the smallest index such that $\sigma(k) \neq \sigma'(k)$. Since σ' is better than σ, we have $\sigma'(k) \succ_1 \sigma(k)$. But then 1 could have picked $\{o_{i_{\sigma(1)}}, \ldots, o_{i_{\sigma(k)}}\}$, thus the condition on Line 5 of Algorithm 3 would have been true, contradicting the fact that Algorithm 3 returns σ. ■

Corollary 2 *Under the lexicographic cardinalization function, the optimal strategy for an agent can be computed in polynomial time.*

Example 5 $n = 4$, $p = 12$, $\pi = 123412341213$;
$\succ_1$: $o_1 \succ o_2 \succ o_3 \succ o_4 \succ o_5 \succ o_6 \succ o_7 \succ o_8 \succ o_9 \succ o_{10} \succ o_{11} \succ o_{12}$
$\succ_2$: $o_{12} \succ o_5 \succ o_8 \succ o_6 \succ o_2 \succ o_7 \succ o_{10} \succ o_9 \succ o_{11} \succ o_1 \succ o_3 \succ o_4$
$\succ_3$: $o_4 \succ o_6 \succ o_3 \succ o_9 \succ o_1 \succ o_7 \succ o_8 \succ o_2 \succ o_5 \succ o_{12} \succ o_{10} \succ o_{11}$
$\succ_4$: $o_7 \succ o_6 \succ o_2 \succ o_5 \succ o_{10} \succ o_8 \succ o_{11} \succ o_1 \succ o_9 \succ o_{12} \succ o_3 \succ o_4$.

- *we check whether there exists a* $\{o_1\}$*-successful strategy (by calling Algorithm 1). It succeeds.* $S \leftarrow \{o_1\}$.
- *we check whether there exists a* $\{o_1, o_2\}$*-successful strategy. It succeeds.* $S \leftarrow \{o_1, o_2\}$.
- *we check whether there exists a* $\{o_1, o_2, o_3\}$*-successful strategy. It succeeds.* $S \leftarrow \{o_1, o_2, o_3\}$.
- *we check whether there exists a* $\{o_1, o_2, o_3, o_4\}$*-successful strategy. It fails.* S *is still* $\{o_1, o_2, o_3\}$.
- *we check whether there exists a* $\{o_1, o_2, o_3, o_5\}$*-successful strategy. It fails.* S *is still* $\{o_1, o_2, o_3\}$.
- *we check whether there exists a* $\{o_1, o_2, o_3, o_6\}$*-successful strategy. It fails.* S *is still* $\{o_1, o_2, o_3\}$.
- *we check whether there exists a* $\{o_1, o_2, o_3, o_7\}$*-successful strategy. It fails.* S *is still* $\{o_1, o_2, o_3\}$.
- *we check whether there exists a* $\{o_1, o_2, o_3, o_8\}$*-successful strategy. It fails.* S *is still* $\{o_1, o_2, o_3\}$.
- *we check whether there exists a* $\{o_1, o_2, o_3, o_9\}$*-successful strategy. It succeeds and returns the strategy* $\sigma(1) = o_2$, $\sigma(2) = o_1$, $\sigma(3) = o_3$, $\sigma(4) = o_9$.

Since 1 gets four objects in π*, it is not possible to do better, therefore the optimal strategy for 1 is* $\sigma(1) = o_2$, $\sigma(2) = o_1$, $\sigma(3) = o_3$, $\sigma(4) = o_9$.

A further question is whethe there exists a cardinalization function for which manipulation is hard (which we believe to be true). A probably more complicated question is whether the manipulation problem is NP-hard for Borda scoring. We conjecture that it is, but we believe that the proof will be hard to find, as it might be related to problem of coalitional unweighted manipulation of voting under the Borda rule, whose complexity is an open problem (see for instance [Xia *et al.*, 2009]).

5 Conclusion

We have defined a generic model of a very intuitive protocol for allocating indivisible goods to agents without eliciting their preferences, and studied it from the points of view of the computation of optimal sequences and the complexity of manipulation by one agent. Further work includes finding the missing complexity results for the FI case, evaluating the probability that the resulting allocation is envy-free, and developing a full game-theoretic analysis of the process.

References

S. Brams and P. Fishburn. Fair division of indivisible items between two people with identical preferences: Envy-freeness, pareto-optimality, and equity. *Social Choice and Welfare*, 17:247–267, 2002.

S. Brams and D. King. Efficient fair division: Help the worst off or avoid envy ? *Rationality and Society*, 17:387–421, 2005.

S. Brams and A. Taylor. *Fair Division — From Cake-cutting to Dispute Resolution*. Cambridge University Press, 1996.

Steven J. Brams and Alan D. Taylor. *The Win-win Solution. Guaranteeing Fair Shares to Everybody*. W. W. Norton & Company, 2000.

S. Brams, M. Kilgour, and C. Klamler. The undercut procedure – an algorithm for the envy-free division of indivisible items. *Social Choice and Welfare*, 2011. To appear.

Y. Chevaleyre, P. Dunne, U. Endriss, J. Lang, M. Lemaître, N. Maudet, J. Padget, S. Phelps, J. Rodríguez-Aguilar, and P. Sousa. Issues in multiagent resource allocation. *Informatica*, 30:3–31, 2006.

Y. Chevaleyre, U. Endriss, and N. Maudet. Simple negotiation schemes for agents with simple preferences: sufficiency, necessity and maximality. *JAAMAS*, 20(2):234–259, 2010.

J. Robertson and W. Webb. *Cake-cutting Algorithms: Be Fair if You Can*. A. K. Peters, Ltd, 1998.

L. Xia, M. Zuckerman, A. Procaccia, V. Conitzer, and J. Rosenschein. Complexity of unweighted coalitional manipulation under some common voting rules. In *IJCAI*, pages 348–353, 2009.