Expected Outcomes and Manipulations in Online Fair Division*

Martin Aleksandrov[1,2,3](✉) and Toby Walsh[1,2,3]

[1] UNSW Sydney, Kensington, Australia
[2] Data61, CSIRO, Melbourne, Australia
{martin.aleksandrov,toby.walsh}@data61.csiro.au
[3] TU Berlin, Berlin, Germany

Abstract. Two simple and attractive mechanisms for the fair division of indivisible goods in an online setting are LIKE and BALANCED LIKE. We study some fundamental computational problems concerning the outcomes of these mechanisms. In particular, we consider what expected outcomes are *possible*, what outcomes are *necessary* and how to compute their *exact* outcomes. In general, we show that such questions are more tractable to compute for LIKE than for BALANCED LIKE. As LIKE is strategy proof but BALANCED LIKE is not, we also consider the computational problem of how, with BALANCED LIKE, an agent can compute a strategic bid to improve their outcome. We prove that this problem is intractable in general.

1 Introduction

Fair division is a fundamental problem in allocating resources among competing agents. Many practical fair division problems are online. We present two such settings. For example, in a food bank, we must start allocating food as it is donated. It is too late to wait until the end of the day before we start distributing the food to charities. As a second example, in allocating deceased organs to patients we must match newly donated organs swiftly. We cannot wait till more organs arrive before deciding on the precise match.

Motivated by such problems, Walsh has proposed a simple online model for the fair division of indivisible items in which the items arrive over time [19]. Aleksandrov et al. analysed two simple and attractive randomized mechanisms for such fair division problems: LIKE and BALANCED LIKE [1]. The LIKE mechanism allocates an arriving item uniformly at random between the agents that "like" it. It satisfies equal treatment of equals, and it is both strategy proof and envy free ex ante [1]. Indeed, any mechanism that is envy free ex ante assigns items to agents with the same probabilities as LIKE does. However, the LIKE mechanism is not very fair ex post as it can possibly allocate all items to one agent. The BALANCED LIKE mechanism is fairer. It allocates an arriving item uniformly at random between the agents that "like" it who have the fewest items currently. BALANCED LIKE bounds the envy one agent has for another's allocation ex post. However, this comes at the price of no longer being strategy proof

G. Kern-Isberner et al. (Eds.): KI 2017, LNAI 10505, pp. 29–43, 2017.
DOI: 10.1007/978-3-319-67190-1_3
* Project funded by the European Research Council (ERC) under the EU's Horizon 2020 research and innovation programme via grant AMPLIFY 670077.

in general [1]. When restricted to 2 agents and 0/1 utilities, Balanced Like is strategy proof. These mechanisms are simple and satisfy many desirable axioms. For these reasons, we now turn attention to their computational properties.

In practice, it may be difficult to query the agents each time an item arrives. The chair will often collect the preferences of the agents in advance, and allocate items to agents as they arrive. There are several settings where it is reasonable to suppose that the chair does that. For instance, in the food bank problem, a good proxy for the utility of an item to a charity that likes it might simply be its retail price. This is public information. As a second example, in deceased organ matching, the utility of allocating an organ to a patient might be computed from a simple formula that takes account of the age of the organ, the age of the patient and a number of other medical factors. This is again public information. The chair might then be interested in what outcomes are possible, necessary or exact based on these declared preferences. For example, the chair might be concerned that agents receive enough utility or particular essential items. Alternatively, the chair might want to be sure that a favored agent gets a particular item. Also, they might even want to give similar utility to each agent or bias the future allocation in case some agents receive only a few items and are promised to receive more in expectation.

There are two sources of uncertainty in deciding these outcomes. First, both mechanisms are randomized. Therefore each mechanism returns a probability distribution over actual outcomes. Second, as the problem is online, the arrival order of items is typically unknown. We consider here the problem of the chair computing what outcomes are possible, necessary or exact depending on both sources of uncertainty. In particular, we focus on computing whether an agent can possibly or necessarily receive a given expected utility. These results easily translate into whether an agent can possibly or necessarily receive a given item. We simply give most of the agent's utility to that item. Also, as all our results hold in the case of binary utilities, they can also be viewed as computing whether an agent can possibly or necessarily receive a given expected number of items. Whilst some of our results consider general utilities, such utilities are mainly used to compare outcomes and do not need to be elicited explicitly. General utilities are not used when bidding or allocating items. Such "like" and "not like" reporting has advantages. It is simple, does not require costly eliciting of utilities of agents for items and it also leads to mechanisms with nice axioms.

Our contributions: We consider three settings: the chair knows the arrival ordering of items, the arrival ordering is drawn from some probability distribution, and the allocation of past items is known. In all settings, we study the problem of the chair computing possible, necessary and exact outcomes of Like and Balanced Like. For both mechanisms, these problems are intractable even with 2 agents and when the ordering of items is not fixed. In contrast, with any number of agents, computing each of these outcomes is tractable for Like and intractable for Balanced Like when the ordering of items is fixed. Interestingly, computing outcomes with Balanced Like becomes tractable in this setting only when restricted to 2 agents. Further, computing outcomes is tractable for both mechanisms at a certain moment of time when a new item arrives supposing the allocation of past items is known. In addition, we study a closely related problem of whether an agent can manipulate these mechanisms by strategically misreporting their preferences. Our computational results have a number of interesting consequences. For example, recall that the Balanced Like mechanism is fairer but not strategy proof. However, we show that computing a manipulation of this mechanism is intractable in general.

2 Preliminaries

We next provide basic definitions of online instances, the Like and Balanced Like mechanisms and their outcomes.

Allocation instance: An *instance* $\mathcal{I} = (A, O, U, \Delta)$ of an online fair division problem has (1) a set A of *agents* $a_1, \ldots, a_n$, (2) a set O of indivisible *items* $o_1, \ldots, o_m$, (3) a matrix $U = (u_{ik})_{m \times n}$ where u_{ik} is the *cardinal utility* of agent a_i for item o_k and (4) a matrix $\Delta = (\delta_{kj})_{m \times m}$ where δ_{kj} is a *probability* that item o_k arrives in moment j.

We consider *binary* utilities and *general* rational non-negative utilities. We say that agent a_i *likes* item o_k if $u_{ik} > 0$. Further, we assume that one item arrives in each moment j, i.e. $\sum_{k=1:m} \delta_{kj} = 1$.

Online setting: Suppose items o_1 to o_j have arrived at moments 1 to j, respectively. Given $o = (o_1, \ldots, o_j)$, let $\Delta(o)$ be its probability, $\pi(j, o)$ the current allocation of these items to agents, $p(\pi(j, o))$ its probability and $u_i(\pi(j, o))$ the additive utility of agent a_i for the items they receive in $\pi(j, o)$. Now, suppose that item o_k arrives at moment $(j+1)$ with probability $\delta_{k(j+1)}$ when each agent a_i places a rational non-negative *bid* v_{ik} for this item and a *mechanism* then decides its allocation to a *feasible* agent in an *online* manner, i.g. given $\pi(j, o)$ and *no* information about future items.

Mechanisms: We consider the randomized Like and Balanced Like mechanisms from [1]. With the Like mechanism, agent a_i is feasible for item o_k if $v_{ik} > 0$. With the Balanced Like mechanism, agent a_i is feasible for item o_k if $v_{ik} > 0$ and have so far received fewest items given $\pi(j, o)$ among those agents that bid positively for item o_k. Let the number of feasible agents be f_k. The probability that a feasible agent a_i is allocated item o_k is equal to $1/f_k$.

Possible, necessary and exact outcomes: We consider *expected* probabilities depending on what information is available to the chair. If the allocation $\pi(j, o)$ is the only available information, we use $p_i(j+1, \pi(j, o))$ for the *probability* of agent a_i for the item that arrives at moment $(j+1)$. If the order o is the only available information, we use $p_i(j+1, o)$ for the *probability* of agent a_i for the item that arrives at moment $(j+1)$. It is equal to $\sum_{\pi(j,o)} p(\pi(j, o)) \cdot p_i(j + 1, \pi(j, o))$. If there is no information about o or $\pi(j, o)$, we use $p_i(j+1)$ for the *probability* of agent a_i for the item that arrives at moment $(j+1)$. It is equal to $\sum_o \Delta(o) \cdot p_i(j+1, o)$. We next define expected utilities of agents for items in each of these settings. Given $\pi(j, o)$, we use $u_{ij}(\pi(j, o))$ for the *utility* of agent a_i. It is equal to $u_i(\pi(j, o)) + p_i(j + 1, \pi(j, o))$. Given o, we use $u_{ij}(o)$ for the *utility* of agent a_i. It is equal to $\sum_{\pi(j,o)} p(\pi(j, o)) \cdot u_i(\pi(j, o))$. Given Δ, we use $u_{ij}(\Delta)$ for the *utility* of agent a_i. It is equal to $\sum_o \Delta(o) \cdot u_{ij}(o)$.

The *probability (or utility)* of agent a_i at moment j is *possible* if their probability (or utility) is positive. The outcome of agent a_i at moment j is *necessary* at least some rational number k if their probability (or utility) is at least k. We also say that the outcome of agent a_i at moment j is *exact* if we want to compute the exact value of their probability (or utility).

We study the complexity of computing possible, necessary and exact outcomes. For a mechanism that allocates all items to agents that like them, note that possible and necessary outcomes are directly related. For this reason, we only study necessary and exact outcomes. Our results for possible outcomes are inherited. We next show this relation.

Suppose we ask if $p_i(j + 1) > 0$ holds. This is true iff there is an ordering o and allocation $\pi(j, o)$ of the first j items such that $p_i(j + 1, \pi(j, o)) > 0$. We therefore conclude that $p_i(j + 1) > 0$ iff $p_i(j + 1) \geq \epsilon$ where $0 < \epsilon \leq \min_{o,\pi(j,o)} \Delta(o) \cdot p(\pi(j, o)) \cdot p_i(j + 1, \pi(j, o))$. Note that this minimum value is positive and, consequently, such ϵ always exists. Such a relation is not true for utilities. For the utility of agent a_i, we have that $u_{ij}(\Delta) > 0$ holds iff agent a_i bids positively for at least one item and at least one item arrives. This problem is easy to decide. However, deciding if $u_{ij}(\Delta) \geq k$ holds might not be so easy.

Recall that we consider three settings: when the past allocation of items to agents is known, when the ordering of items is unknown and when the ordering of items is known. We next observe that all outcomes are tractable in the setting when the past allocation is known, *fixed* and *no* information about future items is available.

Items arriving online: Let us suppose that the first j items have arrived and their allocation be $\pi(j, o)$. Suppose now that item o_k arrives at moment $(j + 1)$. For both Like and Balanced Like, the *exact* value of $p_i(j+1, \pi(j, o))$ is equal to $\sum_{k=1:m} \delta_{k(j+1)} \cdot (1/f_k)$ and the *exact* value of $u_i(\pi(j, o))$ is equal to the sum of the cardinal utilities of agent a_i for the items they are allocated in $\pi(j, o)$. Both of these exact outcomes, the value of $u_{ij}(\pi(j, o))$ and therefore any *possible* and *necessary* outcomes in this setting can be computed in $\mathcal{O}(m \cdot n)$ time and space.

We use popular reductions and computational problems from computational complexity, graph theory and set theory in order to show our hardness results.

Computational complexity: We use complexity classes of decision and counting problems such as P, NP, coNP and #P, and mappings such as *Karp*, *Turing*, *parsimonious* and *arithmetic* reductions [7,16,17].

Graph theory: Let G be an undirected bipartite graph. A *matching* μ in G is a set of vertex-disjoint edges. We say that μ *matches* a vertex if there is an edge in it that is incident with the vertex. Matching μ is *maximal* if it is no longer a matching once some other edge is added to it. Matching μ is *perfect* if it matches all vertices in G. Given a graph G and a number k, the *minimum size maximal matching problem* is to decide if there is a matching μ in G with $|\mu| \leq k$. It is NP-hard on various bipartite graphs [9,15]. Given a graph G, the *counting perfect matchings problem* is to output the number of perfect matchings in G. It is #P-hard on various bipartite graphs [14,18].

Set theory: Let S be a set of integers and b, c be integers. A (b, c)-subset of S is a subset of S whose elements sum up to b and its cardinality is c. The *(b, c)-subset sum problem* is to decide if there is a (b, c)-subset of S. Note that there is a (b, c)-subset of S for at least one $c \in [1, |S|]$ iff there is a subset of S whose elements sum up to b. The latter problem is the NP-hard *b-subset sum problem* [11].

This paper is structured as follows. In Sect. 3, the items are drawn from some known probabilistic distribution Δ. For example, such distribution in the food bank problem could be estimated based on historical data. In Sect. 4, we suppose the ordering o in which the items will arrive is fixed, i.e. for each moment j, we have that $\delta_{kj} = 1$ holds for exactly one item o_k. Again, in the food bank problem, some charities donate certain items on a regular basis and only at specific moments. In Sect. 5, we consider problems of computing manipulations of these mechanisms.

3 Items Arriving from a Distribution

We suppose the agents act sincerely and begin with the case when the chair knows the utilities *but* the items come from a distribution Δ whose size is polynomial in n and m.

StochasticExactUtility	StochasticNecessaryUtility
Input: $\mathcal{I} = (A, O, U, \Delta)$, a_i.	Input: $\mathcal{I} = (A, O, U, \Delta)$, a_i, $k \in \mathbb{Q}$.
Output: $u_{im}(\Delta)$.	Question: $u_{im}(\Delta) \geq k$?

The stochastic exact outcomes of Like and Balanced Like are #P-hard with just two agents. Our reduction is motivated by the food bank problem. Let m items be donated by m suppliers and not each of the suppliers can donate each of the items. This relation could be viewed as an undirected bipartite graph. The items are in one partition. The suppliers are in another partition. Let us enumerate them from 1 to m. There is an edge between an item and a supplier if the supplier donates the item. Each perfect matching in the graph then can be viewed as an ordering w.r.t. the enumeration of the suppliers in which each of the m different suppliers donates exactly one of the m different items. At the beginning of the day, the chair does not know the actual order in which the suppliers will donate items but they can estimate it by computing an estimate δ_{kj} for each item o_k and moment j. Based on past data whose size is polynomial in m, one such estimate could be the number of days of past data in which each of the m items is donated from a different supplier amongst the m suppliers divided by the total number of days of past data. We give a reduction from the *counting perfect matchings problem* to StochasticExactUtility.

Reduction 1. Let G be a (3-regular) bipartite graph with M vertices in each partition. The allocation instance $\mathcal{I}_G$ has:

- **Agents:** agents a_1 and a_2 (i.e. 2 agents),
- **Items:** items o_1 to o_M (i.e. M items),
- **Utilities:** $u_{ij} = 1$ for each a_i and o_j, and
- **Distribution:** $\delta_{kj} = 1/M$ for each o_k and j.

Theorem 1. *With $n = 2$ agents, 0/1 utilities and the* Like *or* Balanced Like *mechanism, problem* StochasticExactUtility *is* #P*-hard under arithmetic reductions.*

Proof. WLOG, the set of orderings of items is equal to the set of perfect matchings in G united with the set of o_ϵ that reveals no items. Each ordering o_M that reveals M items corresponds to a perfect matching in G w.r.t. the enumeration of the suppliers in G. We suppose the items arrive independently of each other and across the different time moments. Consequently, ordering o_M occurs with probability $1/M^M$ and the expected utility $u_{iM}(o_M)$ is $M/2$ with both mechanisms as both agents have the same utilities for items. The ordering o_ϵ reveals 0 items. It occurs with probability 1 minus $(1/M^M)$ multiplied by the number of perfect matchings in G and $u_{i0}(o_\epsilon)$ is 0 with both mechanisms as no items are revealed. We quickly obtain that $u_{iM}(\Delta)$ is equal to $(1/M^M) \cdot (M/2)$ multiplied by the number of perfect matchings in G. The result follows. □

We further showed that stochastic necessary outcomes of these mechanisms are NP-hard with just two agents. We omit the complete proof for reasons of space but we give the main reduction which is from the (b, c)*-subset sum problem*. Given set of integers $S = \{n_1, \ldots, n_M\}$ and integers b and c, we construct instance $\mathcal{I}_{S,b,c}$: (1) agents a_1 and a_2, (2) item o_k for each $n_k \in S$, (3) agent a_i values item o_k with n_k, and (4) $\delta_{kj} = 1/M$ for each item o_k and moment j. The instance of StochasticNecessaryUtility has $\mathcal{I}_{S,b,c}$, agent a_i and constant $k = (1/M^c) \cdot (b/2)$. Let us order each subset of S w.r.t. the enumeration $(1, \ldots, M)$. The set of orderings is now equal to the set of ordered (b, c)-subsets of S united with the set of o_ϵ that reveals no items. Similarly to the proof of Theorem 1, it should be easy now for the reader to show that there is a (b, c)-subset of S iff $u_{iM}(\Delta) \geq k$.

4 Items Arriving from a Fixed Ordering

We again suppose the agents act sincerely and next consider the case that the chair knows the utilities *and* the arrival ordering of future items. This corresponds to the case when exactly one item arrives with probability of one at each moment in time.

ExactUtility	NecessaryUtility
Input: $\mathcal{I} = (A, O, U, o)$, a_i. Output: $u_{im}(o)$.	Input: $\mathcal{I} = (A, O, U, o)$, a_i, $k \in \mathbb{Q}$. Question: $u_{im}(o) \geq k$?

4.1 The Case of $n > 2$ Agents

Let there be $n > 2$ agents. Interestingly, the outcomes of the Like mechanism become tractable whereas the ones of the Balanced Like mechanism remain intractable even when the ordering is fixed.

Exact Outcomes. Let us start with the Like mechanism. This mechanism does not keep track of the allocation of past items. As a result, any agent is feasible for each next item supposing they like this item. Indeed, all exact outcomes are tractable with this mechanism for this reason.

Observation 1. *With general utilities and the* Like *mechanism, problem* ExactUtility *is in* P.

Proof. The probability $p_i(j, o)$ of agent a_i for item o_j is $1/n_j$ where n_j is the number of agents that like the item. Their utility $u_{im}(o)$ can be given as $\sum_{j=1}^{m}(1/n_j) \cdot u_{ij}$. □

We continue with exact allocations for the Balanced Like mechanism and give a parsimonious reduction from *counting perfect matchings problem* to ExactUtility. The counting problem remains in #P-hard even on 3-regular undirected bipartite graphs in [8]. Our reduction is very insightful because it provides a very tight bound on the complexity of ExactUtility (i.e. 0/1 utilities, each agent likes at most 4 items, each item except one is liked by at most 3 agents, each pair of agents like at most 3 items in common, the ordering is fixed, etc.).

Reduction 2. Let G be a 3-regular bipartite graph, $u_1, \ldots, u_N$ be the vertices from one of its partitions and $v_1, \ldots, v_N$ the vertices from the other one of its partitions. For each vertex u_i, let v_{i1}, v_{i2}, v_{i3} denote the vertices connected to it and $e_{3\cdot(i-1)+1} = (u_i, v_{i1})$, $e_{3\cdot(i-1)+2} = (u_i, v_{i2})$, $e_{3\cdot(i-1)+3} = (u_i, v_{i3})$ the edges incident with it. Each edge e_k can be represented as (u_i, v_j) for some $u_i \in \{u_1, \ldots, u_N\}$ and $v_j \in \{v_{i1}, v_{i2}, v_{i3}\}$. We use the graph and next construct the online allocation instance $\mathcal{E}_G$ as follows:

- **Agents:** 1 agent a_k per edge e_k and 3 special agents $a_{3\cdot N+1}$, $a_{3\cdot N+2}$ and $a_{3\cdot N+3}$ (i.e. $3 \cdot N + 1$ agents),
- **Items:** 1 item per vertex v_j, 2 items u_{i1}, u_{i2} per vertex u_i and 3 special items w and x (i.e. $3 \cdot N + 2$ items),
- **Non-zero utilities:** for $i \in [1, N], j \in \{1, 2, 3\}$, **agent** $a_{3\cdot(i-1)+j}$ has utility 1 for items $v_{ij}, u_{i1}, u_{i2}, x$; **agent** $a_{3\cdot N+1}$ has utility 1 for items w, x, and
- **Ordering:** $o = (v_1 \ldots v_N u_{11} u_{12} \ldots u_{N1} u_{N2} w x)$.

We highlight the main idea behind the proof of the next Lemma 1. Basically, we showed that computing the number of allocations of the first $3 \cdot N + 1$ items in o in which each agent receives exactly one item is in #P-complete.

Lemma 1. *With the* BALANCED LIKE *mechanism, the number of allocations in $\mathcal{E}_G$ in which agent $a_{3 \cdot N+1}$ is feasible for item x is equal to 2^N times the number of perfect matchings in G. Computing it is in #P-hard under arithmetic reductions.*

Proof. By construction, each item v_j is liked by three different agents and, hence, each allocation of $v_1, \ldots, v_N$ gives these items to N different agents among $a_1, \ldots, a_{3 \cdot N}$. Consider then an allocation of $v_1, \ldots, v_N$ such that, for each vertex u_i, either agent $a_{3 \cdot (i-1)+1}$ gets item v_{i1} or agent $a_{3 \cdot (i-1)+2}$ gets item v_{i2} or agent $a_{3 \cdot (i-1)+3}$ gets item v_{i3}. We say that such an allocation of $v_1, \ldots, v_N$ has *perfect* matches for vertices $u_1, \ldots, u_N$ because exactly one agent per triplet $a_{3 \cdot (i-1)\,+\,1}, a_{3 \cdot (i-1)\,+\,2}, a_{3 \cdot (i-1)\,+\,3}$ gets an item among $v_1, \ldots, v_N$. In fact, there is a perfect matching in G over $v_1, \ldots, v_N$ and $u_1, \ldots, u_N$ iff there is an allocation in $\mathcal{E}_G$ of $v_1, \ldots, v_N$ that has perfect matches for $u_1, \ldots, u_N$. Furthermore, this is a 1-to-1 parsimonious correspondence. Each allocation π in $\mathcal{E}_G$ of the first $3 \cdot N+1$ items in o in which each agent among $a_1, \ldots, a_{3 \cdot N}, a_{3 \cdot N+1}$ receives exactly one item occurs with positive probability. We call π *perfect allocation* over the first $3 \cdot N + 1$ items in o. We show that there is an allocation in $\mathcal{E}_G$ of $v_1, \ldots, v_N$ that has perfect matches for $u_1, \ldots, u_N$ iff there are 2^N perfect allocations such as π in $\mathcal{E}_G$. Moreover, this is a 1-to-2^N arithmetic correspondence. In other words, we show that the number of perfect allocations such as π in $\mathcal{E}_G$ is equal to 2^N times the number of perfect matchings in G.

First, let us consider one discrete allocation π_1 in $\mathcal{E}_G$ of $v_1, \ldots, v_N$ that has perfect matches for $u_1, \ldots, u_N$. The allocation π_1 occurs with positive probability because $v_1, \ldots, v_N$ are liked by disjoint sets of three agents. WLOG, suppose that π_1 is such that, for each u_i, agent $a_{3 \cdot (i-1)\,+\,1}$ receives their corresponding item v_{i1}. The allocation π_1 can be extended by the mechanism to two discrete allocations w.r.t. each u_i: (1) agent $a_{3 \cdot (i-1)\,+\,2}$ gets item u_{i1} and agent $a_{3 \cdot (i-1)\,+\,3}$ gets item u_{i2} or (2) agent $a_{3 \cdot (i-1)\,+\,2}$ gets item u_{i2} and agent $a_{3 \cdot (i-1)\,+\,3}$ gets item u_{i1}. By the preference structure, π_1 can then be extended by the mechanism to 2^N perfect allocations in $\mathcal{E}_G$. Note that each of these perfect allocations necessarily gives item w to agent $a_{3 \cdot N+1}$ because only they like it. Second, consider one perfect allocation in $\mathcal{E}_G$. It must be the case that it extends some discrete allocation of $v_1, \ldots, v_N$ that has perfect matches for $u_1, \ldots, u_N$. To show this, consider a discrete allocation π_2 of $v_1, \ldots, v_N$ that has not perfect matches for $u_1, \ldots, u_N$. Hence, π_2 is such that at least two of the agents $a_{3 \cdot (i-1)\,+\,1}, a_{3 \cdot (i-1)+2}, a_{3 \cdot (i-1)\,+\,3}$ for some vertex u_i receive their corresponding items v_{i1}, v_{i2}, v_{i3} of $v_1, \ldots, v_N$. Therefore, each allocation of all items that extends π_2 by using the mechanism gives item u_{i1} or item u_{i2} to one of the agents $a_{3 \cdot (i-1)\,+\,1}, a_{3 \cdot (i-1)\,+\,2}, a_{3 \cdot (i-1)\,+\,3}$ as their second item. As a consequence, in each such allocation, there is another agent with zero items after round $3 \cdot N+1$. We conclude that each such extension of π_2 is not a perfect allocation in $\mathcal{E}_G$. $\square$

Theorem 2. *With $n > 2$ agents, 0/1 utilities and the* BALANCED LIKE *mechanism, problem* EXACTUTILITY *is in #*P*-hard under arithmetic reductions.*

Proof. Let us consider allocation $\pi = \pi(3 \cdot N + 1, o)$ of the first $3 \cdot N + 1$ items in o in which each agent among $a_1, \ldots, a_{3 \cdot N}, a_{3 \cdot N+1}$ receives exactly one item. Note that agent $a_{3 \cdot N+1}$ gets item x with positive conditional probability only given such allocations because all agents like item x. By the preference structure, we conclude that π occurs with probability $p(\pi) = (1/3^N) \cdot (1/2^N)$. The conditional probability $p_i(x|\pi)$ of agent $a_{3 \cdot N+1}$ for item x given π is equal to $1/(3 \cdot N + 1)$ because all agents $a_1, \ldots, a_{3 \cdot N}, a_{3 \cdot N+1}$ like item x. The conditional probability of agent $a_{3 \cdot N+1}$ for item x is 0 given any other allocation. Therefore, $p_{3 \cdot N+1}(x, o)$ is equal to $(1/3^N) \cdot (1/2^N) \cdot (1/(3 \cdot N + 1))$ multiplied by the number of allocations such as π in which agent $a_{3 \cdot N+1}$ is feasible for item x. Finally, the expected utility $u_{(3 \cdot N+1)(3 \cdot N+3)}(o) = p_{3 \cdot N+1}(w, o) + p_{3 \cdot N+1}(x, o)$. We have that $p_{3 \cdot N+1}(w, o) = 1$ because only agent $a_{3 \cdot N+1}$ likes item w and the mechanism allocates each item to an agent. The result follows by Lemma 1. □

Necessary Outcomes. The tractability of the exact allocations of the LIKE mechanism entails the tractability of its necessary allocations. By Observation 1, we conclude the next immediate result.

Observation 2. *With general utilities and the* LIKE *mechanism, problem* NECESSARYUTILITY *is in* P.

We next focus on the necessary outcomes of the BALANCED LIKE mechanism. We give a Karp reduction from *minimum size maximal matching problem* to the negation of NECESSARYUTILITY. The minimum size maximal matching problem is shown to be NP-hard on subdivision graphs of degree at most 3 in [12].

Reduction 3. Let us have a subdivision graph G of degree at most 3 and integer r. The graph G is bipartite with vertices $u_1, \ldots, u_N$ of degree exactly 2 and vertices $v_1, \ldots, v_M$ of degree at most 3. WLOG, we can assume that $N \geq M$ and there are no two vertices from U that are connected to the same two vertices from V. We construct an allocation instance $\mathcal{P}_{G,r}$ as follows:

– **Agents:** 2 agents u_{i1}, u_{i2} per u_i and agents $a_1, \ldots, a_{N-r}$, $b_1, \ldots, b_M$ and c (i.e. $3 \cdot N + M - r + 1$ agents),
– **Items:** 1 item per v_j and items $x_1, \ldots, x_N$, $y_1, \ldots, y_N$, $z_1, \ldots, z_{N-r}$ and w (i.e. $3 \cdot N + M - r + 1$ items),
– **Non-zero utilities:** for each $i \in [1, N], j \in \{1, 2\}$, **agent** u_{ij} has utility 1 for items $x_i, v_{ij}, y_i, z_1, \ldots, z_{N-r}$; for each $i \in [1, N - r]$, **agent** a_i has utility 1 for items $x_1, \ldots, x_N$; **agents** $b_1, \ldots, b_M$ have each utility 1 for item w; **agent** c has utility 1 for items z_{N-r}, w, and
– **Ordering:** $o = (x_1 \ldots x_N v_1 \ldots v_M y_1 \ldots y_N z_1 \ldots z_{N-r} w)$.

The expected utility of each of the agents $b_1, \ldots, b_M$ is at least $1/M$ iff $p_c(w, o) = 0$. This observation holds because each of the agents b_1 to b_M have equal utilities for items in which case they receive item w with the same probability which apparently is also equal to their expected utility as this is the only item they like. Theorem 3 follows from this observation.

Theorem 3. *With $n > 2$ agents, 0/1 utilities and the* Balanced Like *mechanism, problem* NecessaryUtility *is in* coNP*-hard under Turing reductions.*

Proof. There is a maximal matching in G of cardinality at most r iff there is an allocation in $\mathcal{P}_{G,r}$ in which agent c receives item w iff $p_c(w, o) > 0$. The second "iff" is trivial. We, therefore, focus on the first "iff". The "only if" direction is easier to show and, for reasons of space, we only show the more difficult "if" direction. Suppose next that π is an allocation of all items in $\mathcal{P}_{G,r}$ in which agent c receives item w.

1. Item w is allocated in π to agent c as their first item. To see this, suppose they also get some items among z_{N-r}. Now, they would not be feasible when item w arrives as agents $b_1, \ldots, b_M$ have zero items in π and the mechanism would have given item w to an agent among $b_1, \ldots, b_M$ and not to agent c.
2. Prior to item w in π, agent c have received zero items. Hence, items $z_1, \ldots, z_{N-r}$ are allocated in π to $N - r$ agents as their first items. By the preferences, these agents are from different pairs among $u_{11}, u_{12}, \ldots, u_{N1}, u_{N2}$ because, for each pair of agents u_{i1}, u_{i2}, either u_{i1} or u_{i2} is forced to get item y_i. WLOG, let us assume that agents $u_{11}, \ldots, u_{(N-r)1}$ get items $z_1, \ldots, z_{N-r}$ in π.
3. Prior to item z_1 in π, agents $u_{11}, \ldots, u_{(N-r)1}$ have zero items. Hence, $N - r$ items among $y_1, \ldots, y_N$ are allocated in π to $u_{12}, \ldots, u_{(N-r)2}$ as their first items. These items are $y_1, \ldots, y_{N-r}$. For i in $[N-r+1, N]$, we note that item y_i is allocated in π to either u_{i1} or u_{i2} as their first or second item.
4. Prior to item y_1 in π, agents $u_{11}, u_{12}, \ldots, u_{(N-r)1}, u_{(N-r)2}$ have zero items. By the preferences, agents $a_1, \ldots, a_{N-r}$ must then receive items $x_1, \ldots, x_{N-r}$ in π. For i in $[N-r+1, N]$, item x_i is allocated in π to either u_{i1} or u_{i2}, say u_{i2}. We conclude that agents $u_{(N-r+1)1}, \ldots, u_{N1}$ have zero items prior to item v_1 in π. Moreover, only agents $u_{(N-r+1)1}, u_{(N-r+1)2}, \ldots, u_{N1}, u_{N2}$ receive items $v_1, \ldots, v_M$ in π. Finally, only $l \leq r$ agents among $u_{(N-r+1)1}, \ldots, u_{N1}$ get items in π among $v_1, \ldots, v_M$ as first items as some of these agents might like the same items among $v_1, \ldots, v_M$. WLOG, let these agents be $u_{(N-l+1)1}, \ldots, u_{N1}$ and they are allocated in π items $v_1, \ldots, v_l$ as first items.

The constructed set $\mu_\pi = \{(u_{N-l+1}, v_1), \ldots, (u_N, v_l)\}$ contains only edges from the graph G which are vertex-disjoint. Therefore, this set is a matching in G. Moreover, the cardinality of this set is l at most r. We next show that μ_π is a maximal matching. For the sake of contradiction, suppose that μ_π remains a matching if we add a new edge to it, say (u, v). The edge (u, v) is vertex-disjoint with the edges in μ_π. This means that vertex u is not among $u_{N-l+1}, \ldots, u_N$ and vertex v is not among $v_1, \ldots, v_l$. Hence, vertex u is among $u_1, \ldots, u_{N-l}$. In the allocation π, agents $u_{11}, u_{12}, \ldots, u_{(N-r)1}, u_{(N-r)2}$ do not receive any items among $v_1, \ldots, v_M$. This implies that all these agents are feasible for the items they like among $v_1, \ldots, v_M$ but they do not get them in π. As agents $u_{(N-l+1)1}, \ldots, u_{N1}$ get items $v_1, \ldots, v_l$ as their first items, we conclude that some agents among $u_{(N-l+1)1}, u_{(N-l+1)2} \ldots, u_{N1}, u_{N2}$ receive items v_{l+1}, $\ldots, v_M$ as their second items. Therefore, it must be the case that all agents $u_{11}, u_{12}, \ldots, u_{(N-r)1}, u_{(N-r)2}$ do not like any item among $v_{l+1}, \ldots, v_M$. Otherwise, the mechanism would allocate some of these items to agents among u_{11}, u_{12}, $\ldots, u_{(N-r)1}, u_{(N-r)2}$. This is just the way in which the mechanism works. And, we reached a contradiction with the existence of the allocation π. Finally, in the graph G, vertices $u_1, \ldots, u_{N-r}$ are connected only to vertices among $v_1, \ldots, v_l$. Hence, v is among $v_1, \ldots, v_l$. This fact contradicts that $\mu_\pi \cup \{(u, v)\}$ is a matching. □

4.2 The Case of 2 Agents

By Observations 1 and 2, the outcomes of Like are tractable. Surprisingly, in contrast to Theorems 1, 2 and 3, the outcomes of Balanced Like become tractable with only two agents and when the ordering of items is fixed.

Theorem 4. *With $n = 2$ agents, general utilities and the* Balanced Like *mechanism, problems* ExactUtility *and* NecessaryUtility *are in* P*.*

Proof. We use a dynamic program. Each state $s = (p, q)$ in it encodes that agent a_1 has p items, agent a_2 has q items, and its probability $p(s)$. By induction, we show that there are at most 2 different states after each allocation round. In the base case, consider round 1. There are at most 2 states after this round depending on whether both a_1 and a_2 or only one of them like the first item. In the hypothesis, consider round j and suppose there are at most two states after round j. In the step case, consider round $j + 1$. Now, there are two cases. In the first one, there is only one state after round j. The result follows by the base case. In the second case, there are two states after round j. Let these be (p, q) and $(p - 1, q + 1)$ where $p + q = j$. If only one agent likes item o_{j+1}, each state transits into a new state and the result follows. If both a_1 and a_2 like item o_{j+1}, we consider four sub-cases depending on the difference $p-q$: (1) (p, q) and $(p-1, q+1)$ for $p-q > 2$, (2) $(q+2, q)$ and $(q+1, q+1)$ for $p-q = 2$, (3) $(q+1, q)$ and $(q, q + 1)$ for $p - q = 1$ and (4) (q, q) and $(q - 1, q + 1)$ for $p - q = 0$. For sub-case (1), each state transits into one new state with the same probability. For sub-case (2), $(q + 2, q)$ transits into $(q + 2, q + 1)$, and $(q + 1, q + 1)$ into $(q + 2, q + 1)$ and $(q + 1, q + 2)$. For sub-case (3), both states transit into the same new state with probability 1. For sub-case (4), (q, q) transits into $(q, q+1)$ and $(q + 1, q)$, and $(q - 1, q + 1)$ into $(q, q + 1)$. We conclude that there are at most two different states after round $j + 1$ in each sub-case.

The probability $p_1(j + 2, o)$ is equal to $\sum_{s_{j+1}} p(s_{j+1}) \cdot p(a_1 \text{ gets } o_{j+2} | s_{j+1})$ where s_{j+1} is such a state after round $j + 1$ in which agent a_1 is feasible for item o_{j+2}. The conditional probability $p(a_1 \text{ gets } o_{j+2} | s_{j+1})$ of agent a_1 for item o_{j+2} is (i) 0 or 1 in sub-case (1), (ii) 0, 1/2 or 1 in sub-case (3) and (iii) the probability of the state in which they are feasible in sub-cases (2) and (4). We can compute the states, their probabilities and hence the probabilities of agents and their utilities in $\mathcal{O}(m)$ space and time. □

5 Manipulations

We next consider how agents can act strategically. The LIKE mechanism is strategy-proof and hence agents have an incentive to bid sincerely for items. In contrast, the BALANCED LIKE mechanism is not strategy-proof and agents can have an incentive to bid strategically for items [1]. We thus focus on strategic misreporting of bids with BALANCED LIKE. In particular, we study the worst case when the utilities and the ordering of the items are known to the misreporting agent. Any complexity results, in this case, provide lower bounds on the complexity in the case of partial or probabilistic information. We formulate the next problems where $u_{im}(v^i, o)$ denotes the utility of agent a_i supposing their bid vector is $v^i = (v_{i1}, \ldots, v_{im})$ and the other agents bid sincerely. Let $u^i = (u_{i1}, \ldots, u_{im})$ denotes their sincere bid vector.

EXACTMANIPULATION	NECESSARYMANIPULATION
Input: $\mathcal{I} = (A, O, U, o)$, a_i, u^i, v^i.	Input: $\mathcal{I}=(A, O, U, o)$, a_i, v^i, u^i, $k \in \mathbb{Q}$.
Output: $u_{im}(v^i, o) - u_{im}(u^i, o)$.	Question: $u_{im}(v^i, o) - u_{im}(u^i, o) \geq k$?

Theorem 5. *With $n > 2$ agents, 0/1 utilities and the* BALANCED LIKE *mechanism, problem* EXACTMANIPULATION *is in #*P*-hard under arithmetic reductions.*

Proof. Consider instance $\mathcal{E}_G$. Let us modify this instance a bit. We add one new item z between items w and x in the ordering o such that only agent $a_{3 \cdot N+1}$ likes z with 1. Let $\mathcal{F}_G$ denote this new instance. Suppose that all agents in $\mathcal{F}_G$ bid sincerely. Thus, agent $a_{3 \cdot N+1}$ receives each of the items w and z each with probability 1 because they are the only agent who likes them. However, they receive item x with probability 0. Therefore, $u_{(3 \cdot N+1)(3 \cdot N+3)}(u^{(3 \cdot N+1)}, o) = 2$. Suppose that all agents in $\mathcal{F}_G$ bid sincerely except agent $a_{3 \cdot N+1}$ who bids strategically 0 for item z. Let $v^{(3 \cdot N+1)}$ be their bidding vector in this case. We can now remove item z because no agent bids positively for it. But, then we obtain instance $\mathcal{E}_G$. By Theorem 2, we have $u_{(3 \cdot N+1)(3 \cdot N+3)}(v^{(3 \cdot N+1)}, o) = 1 + p_{3 \cdot N+1}(x, o)$. The instance of EXACTMANIPULATION uses instance $\mathcal{F}_G$, agent $a_{3 \cdot N+1}$ and vectors $u^{(3 \cdot N+1)}$ and $v^{(3 \cdot N+1)}$. Its hardness follows by Theorem 2. □

Observe that the truthful report of agent $a_{3 \cdot N+1}$ in the proof of Theorem 5 leads to their utility being 2 whereas their insincere report leads to their utility being at most 2. Hence, their strategic move cannot lead to an increase in their utility but the computation of the exact difference in utility is intractable. However, as we discuss next, computing an exact profitable insincere report that leads to such an increase is also intractable.

Necessary manipulations might be easy even when exact manipulations are hard. For example, in the proof of Theorem 5, suppose that agent $a_{3 \cdot N+1}$ has cardinal utility for item x that is strictly greater than $(3^N).(3N+1)$. If they bid sincerely, their expected utility is 2. If they bid strategically zero for item z, their expected utility is strictly greater than 2. This *necessary* increase can be decided in polynomial time but computing the *exact* increase is intractable. However, necessary manipulations are also in general not always easy even if we ask merely for any increase in the expected utility of a given agent.

Theorem 6. *With $n > 2$ agents, 0/1 utilities and the* BALANCED LIKE *mechanism, problem* NECESSARYMANIPULATION *is in* coNP*-hard under Turing reductions.*

Proof. Consider instance $\mathcal{P}_{G,r}$. Suppose all agents bid sincerely. Hence, $u_{c(3N+M-r+1)}(u^c, o) = p_c(z_{N-r}, o) + p_c(w, o)$. Suppose all agents bid sincerely except agent c who bids strategically 0 for item w. Let their bidding vector be v^c. We have that $u_{c(3N+M-r+1)}(v^c, o) = p_c(z_{N-r}, o)$. The instance of NECESSARYMANIPULATION uses as input instance $\mathcal{P}_{G,r}$, agent c, vectors v^c and u^c, and rational number $k = 0$. We conclude that $u_{c(3N+M-r+1)}(v^c, o) - u_{c(3N+M-r+1)}(u^c, o) \geq 0$ iff $p_c(w, o) = 0$. The result follows by Theorem 3. □

Another definition of the manipulation problem is whether a player can possibly increase their utility by insincere reporting, rather than computing the necessary or exact gain. Observe that in the proof of Theorem 6, we have that $u_{c(3N+M-r+1)}(u^c, o) - u_{c(3N+M-r+1)}(v^c, o) > 0$ iff $p_c(w, o) > 0$. We conclude that possible manipulations are also intractable in general by the proof of Theorem 3. Finally, by Theorem 4, we conclude that possible, necessary and exact manipulations are easy with just two agents and items arriving from a fixed ordering. By Theorem 1 and the discussion after it, we conclude that necessary and exact manipulations are hard with two agents and items arriving from a distribution.

6 Related Work and Conclusion

We studied the worst-case computational complexity of possible, necessary and exact outcomes returned by the LIKE and BALANCED LIKE mechanisms supposing agents act sincerely. With LIKE, there is no benefit for agents to act strategically. With BALANCED LIKE, the agents might be strategic but we proved that computing a manipulation is computationally intractable in general. Some results are however tractable for the case of 2 agents. Our study of the online allocations returned by the LIKE and BALANCED LIKE mechanisms is in-line with many results in offline fair division, voting theory and partial tournaments where possible, necessary and exact outcomes play crucial role; see e.g. [2,4,5,20]. Our results provide a stepping stone towards better understanding strategic behavior. A number of works already considered such behavior for offline mechanisms; see e.g. [3,6]. Another interesting future directions would be to estimate the outcomes of our mechanisms or to look at fixed-parameter tractable algorithms for these problems [10,13,15].

References

1. Aleksandrov, M., Aziz, H., Gaspers, S., Walsh, T.: Online fair division: analysing a food bank problem. In: Proceedings of IJCAI 2015, Buenos Aires, Argentina, pp. 2540–2546, 25–31 July 2015. http://ijcai.org/papers15/Abstracts/IJCAI15-360.html
2. Aziz, H., Brill, M., Fischer, F.A., Harrenstein, P., Lang, J., Seedig, H.G.: Possible and necessary winners of partial tournaments. J. Artif. Intell. Res. (JAIR) **54**, 493–534 (2015). http://dx.doi.org/10.1613/jair.4856
3. Aziz, H., Gaspers, S., Mackenzie, S., Mattei, N., Narodytska, N., Walsh, T.: Equilibria under the probabilistic serial rule. In: Proceedings of IJCAI 2015, Buenos Aires, Argentina, pp. 1105–1112, 25–31 July 2015. http://ijcai.org/papers15/Abstracts/IJCAI15-160.html
4. Aziz, H., Walsh, T., Xia, L.: Possible and necessary allocations via sequential mechanisms. In: Proceedings of IJCAI 2015, Buenos Aires, Argentina, pp. 468–474, 25–31 July 2015. http://ijcai.org/papers15/Abstracts/IJCAI15-072.html
5. Bachrach, Y., Betzler, N., Faliszewski, P.: Probabilistic possible winner determination. In: Proceedings of AAAI 2010, Atlanta, Georgia, USA, 11–15 July 2010
6. Bouveret, S., Lang, J.: Manipulating picking sequences. In: ECAI 2014–21st European Conference on Artificial Intelligence, Prague, Czech Republic - Including PAIS, pp. 141–146, 18–22 August 2014. http://dx.doi.org/10.3233/978-1-61499-419-0-141
7. Bürgisser, P.: Completeness and Reduction in Algebraic Complexity Theory. Algorithms and Computation in Mathematics. Springer, New York (2000). http://opac.inria.fr/record=b1099577
8. Dagum, P., Luby, M.: Approximating the permanent of graphs with large factors. Theor. Comput. Sci. **102**(2), 283–305 (1992)
9. Demange, M., Ekim, T.: Minimum maximal matching is NP-hard in regular bipartite graphs. In: Agrawal, M., Du, D., Duan, Z., Li, A. (eds.) TAMC 2008. LNCS, vol. 4978, pp. 364–374. Springer, Heidelberg (2008). doi:10.1007/978-3-540-79228-4_32
10. Downey, R.G., Fellows, M.R.: Fundamentals of Parameterized Complexity. Texts in Computer Science. Springer, London (2013). https://doi.org/10.1007/978-1-4471-5559-1
11. Garey, M.R., Johnson, D.S.: Computers and Intractability: A Guide to the Theory of NP-Completeness. W. H. Freeman, New York (1979)
12. Horton, J.D., Kilakos, K.: Minimum edge dominating sets. SIAM J. Discrete Math. **6**(3), 375–387 (1993). http://dx.doi.org/10.1137/0406030
13. Jerrum, M., Sinclair, A.: Approximating the permanent. SIAM J. of Comp. **18**(6), 1149–1178 (1989). http://dx.doi.org/10.1137/0218077
14. Okamoto, Y., Uehara, R., Uno, T.: Counting the number of matchings in chordal and chordal bipartite graph classes. In: Paul, C., Habib, M. (eds.) WG 2009. LNCS, vol. 5911, pp. 296–307. Springer, Heidelberg (2010). doi:10.1007/978-3-642-11409-0_26
15. Sabán, D., Sethuraman, J.: The complexity of computing the random priority allocation matrix. Math. Oper. Res. **40**(4), 1005–1014 (2015)
16. Turing, A.M.: On computable numbers, with an application to the Entscheidungsproblem. J. Proc. Lond. Math. Soc. **2**(42), 230–265 (1936). http://www.cs.helsinki.fi/u/gionis/cc05/OnComputableNumbers.pdf

17. Valiant, L.G.: The complexity of computing the permanent. Theor. Comput. Sci. **8**, 189–201 (1979). http://dx.doi.org/10.1016/0304-3975(79)90044-6
18. Valiant, L.G.: The complexity of enumeration and reliability problems. SIAM J. Comput. **8**(3), 410–421 (1979). http://dx.doi.org/10.1137/0208032
19. Walsh, T.: Challenges in resource and cost allocation. In: Proceedings of 29th AAAI 2015, Austin, Texas, USA, pp. 4073–4077, 25–30 January 2015. http://www.aaai.org/ocs/index.php/AAAI/AAAI15/paper/view/9927
20. Xia, L., Conitzer, V.: Determining possible and necessary winners given partial orders. J. Artif. Intell. Res. (JAIR) **41**, 25–67 (2011)