Strategic Behavior when Allocating Indivisible Goods Sequentially

Thomas Kalinowski
University of Rostock
Germany

Nina Narodytska and **Toby Walsh**
NICTA and UNSW
Australia

Lirong Xia
CRCS, Harvard University
USA

Abstract

We study a simple sequential allocation mechanism for allocating indivisible goods between agents in which agents take turns to pick items. We focus on agents behaving strategically. We view the allocation procedure as a finite repeated game with perfect information. We show that with just two agents, we can compute the unique subgame perfect Nash equilibrium in linear time. With more agents, computing the subgame perfect Nash equilibria is more difficult. There can be an exponential number of equilibria and computing even one of them is PSPACE-hard. We identify a special case, when agents value many of the items identically, where we can efficiently compute the subgame perfect Nash equilibria. We also consider the effect of externalities and modifications to the mechanism that make it strategy proof.

Introduction

There are many situations where we need to divide items between agents. We might wish to divide time windows on a shared resources like a telescope, places on courses between students, or players between teams. There are a variety of mechanisms used to do such division without side payments (Brams and Fishburn 2000; Brams, Edelman, and Fishburn 2003; Othman, Sandholm, and Budish 2010; Budish et al. 2013). For example, the Harvard Business School uses the *Draft* mechanism to allocate courses to students (Budish and Cantillon 2007). This mechanism generates a priority order over all students uniformly at random. Courses are then allocated to students in rounds. In odd rounds, each student is assigned to their favorite course that still has availability using the priority order. In even rounds, we reverse the priority order. Unfortunately this mechanism is not strategy-proof, and students have been observed to behave strategically (Budish and Cantillon 2007).

Bouveret and Lang (2011) consider computational aspects of a sequential allocation procedure (studied earlier by Brams and others) which generalizes many aspects of the Draft mechanism. The procedure is parameterized by a policy, the sequence in which agents take turns to pick items. For example, as in the Draft, with two agents and four items, the policy 1221 gives first and last pick to the first agent, and second and third pick to the second agent. One good feature of this mechanism is that the preferences of the agents do not need to be elicited. Bouveret and Lang assume agents have additive utilities given by a common scoring function (e.g. Borda or lexicographic scores). When agents have the same preference ordering, all policies give the same expected sum of utilities, and the mechanism is strategy proof. When agents have different preference orderings, strategic behavior can be profitable. The goal of this paper is to study computational aspects of such strategic behavior.

Formally, we have a set G of m items which we are dividing between n agents. Agent i represents her preferences by the linear order P_i over G. A *policy* $O = o_1 o_2 \cdots o_m \in \{1, \dots, n\}^m$ defines an allocation mechanism in whose ith stage, agent o_i picks an item. With two agents, we write $allocate(P_1, P_2, O)$ for the allocation constructed with the policy O, given the preference orderings P_1 and P_2 supposing agents are truthful. That is, in each stage an agent chooses her most-preferred item still available. We write $rev(P_1)$ for the reverse of the preferences P_1, and $rev(O)$ for the reverse of the allocation order in policy O.

Example 1 *Let $G = \{1, \dots, 6\}$ and $O = 121221$. Agent 1's preferences are $P_1 = 1 > 2 > 3 > 4 > 5 > 6$. Agent 2's preferences are $P_2 = 4 > 6 > 2 > 1 > 5 > 3$. The item allocated in each round of $allocate(P_1, P_2, O)$ is illustrated in the following table.*

O	1	2	1	2	2	1
Item	1	4	2	6	5	3

Strategic behavior

Bouveret and Lang (2011) consider situations where one agent attempts to secure a better allocation by picking strategically supposing all other agents pick sincerely. With lexicographical scores, they show that the optimal strategy for an agent given a particular policy can be computed in polynomial time supposing other agents pick sincerely. They conjecture that finding the optimal strategy is NP-hard for Borda scores. Supposing all agents but the manipulator pick sincerely is a strong assumption. If one agent is picking strategically, why not the others? The sequential allocation procedure naturally lends itself to a game theoretic analysis where we look for a Nash equilibrium where no agent can improve their allocation by deviating unilaterally from their (perhaps insincere) picking strategy. We view the allocation procedure as a finite repeated sequential game in which all agents have complete information about the preference ordering of the other agents. When a policy has an agent picking multiple items in turn, we consider this as one move (so that if an agent picks b then a but prefers a to b then this is not insincere as it the same as the sincere picking a then b).

We can use backward induction to find the subgame perfect Nash equilibrium (SPNE). When agents have the same preference ordering, Proposition 6 in (Bouveret and Lang 2011) proves that this is sincere picking. On the other hand, when preference orderings are different, there exist equilibria where behaviour is not sincere.

Example 2 *Consider 4 items and the alternating policy 1212. Suppose the first agent has the preference order* $1 > 2 > 3 > 4$ *whilst the second agent has* $4 > 2 > 3 > 1$. *Then picking sincerely allocates* 1 *and* 2 *to the first agent and* 4 *and* 3 *to the second. However, the second agent can exploit the sincerity of the first and manipulate the mechanism to get a better allocation. Suppose the first agent sincerely picks* 1, *but the second insincerely picks* 2. *Now the first agent cannot do better than pick* 3, *leaving the second agent with* 4. *In this way, the second agent gets a better allocation (*4 *and* 2*) but the first gets a worse allocation (*1 *and* 3*).*

To prevent this, the first agent can themselves pick strategically by insincerely first picking 2. *The worst possible move for the second agent now is to pick* 1. *Whichever other item the second agent picks (*3 *or* 4*), the first agent will pick their more preferred item* 1. *Hence, the first agent ends up with* 1 *and* 2*, which was the same final allocation as the sincere case. However, by picking insincerely in this way, the first agent prevents the second agent from manipulating the result and worsening their allocation. This insincere strategy for the first agent (picking* 2 *then* 1*) and sincere strategy for the second agent (picking* 4 *then* 3*) is the SPNE found by backward induction.*

Two agents

With two agents, additive utilities, and the strictly alternating policy, Kohler and Chandraesekaran (1971) prove that the subgame perfect Nash equilibrium can be computed in linear time by simply reversing the policy and preference orderings. We have the following surprising extension: for two agents and *any* policy (not just strict alternation), an SPNE can be computed in polynomial time, and this is unique provided no agent has the same utility for any pair of items.

Theorem 1 *With two agents and additive utilities, for any policy* O*,* $allocate(rev(P_2), rev(P_1), rev(O))$ *is an SPNE. Moreover, the allocation in all SPNE is unique if both agents have strict preferences over items.*

Proof: By induction on the number of items m. The case $m = 1$ is trivial. Suppose the theorem holds for $m-1$ items. We show that it holds for m. Let $O = o_1 o_2 \cdots o_m$, $G = \{1, \ldots, m\}$, and $(Q_1, Q_2) = \big((x_1, \ldots, x_{m_1}), (y_1, \ldots, y_{m_2})\big)$ denote the output of $allocate(rev(P_2), rev(P_1), rev(O))$, where for any $i \leq m_1$ (respectively, $i' \leq m_2$), $x_i \in G$ (respectively, $y_{i'} \in G$) is the (m_1+1-i)th (respectively, (m_2+1-i')th) item allocated to agent 1 (respectively, agent 2) using $rev(O)$.

W.l.o.g. $o_1 = 1$. Now let us focus on any SPNE of the game (P_1, P_2, π), where agent 1 chose an item in the first round. If she chooses x_1, then by induction hypothesis, $\big(\{x_2, \ldots, x_{m_1}\}, \{y_1, \ldots, y_{m_2}\}\big)$ is the unique SPNE allocation for $G \setminus \{x_1\}$ according to $o_2 \to \cdots \to o_m$, which prove the case for m. We next show that agent 1 has no incentive to choose any other item in the first stage of (P_1, P_2, O).

For the sake of contradiction, suppose agent 1 obtains a higher utility in an SPNE $\big((x_1^*, \ldots, x_{m_1}^*), (y_1^*, \ldots, y_{m_2}^*)\big)$, where for any $i \leq m_1$ (respectively, $i' \leq m_2$), $x_i^* \in G$ (respectively, $y_{i'} \in G$) is the item that agent 1 (respectively, agent 2) chooses in the ith (respectively, i'th) time of her play, and $x_1^* \neq x_1$. We next show a contradiction by proving that $u_1(\{x_1, \ldots, x_{m_1}\}) \geq u_1(\{x_1^*, \ldots, x_{m_1}^*\})$, and the equality holds if and only if $\{x_1, \ldots, x_{m_1}\} = \{x_1^*, \ldots, x_{m_1}^*\}$ (alternatively, $\{y_1, \ldots, y_{m_1}\} = \{y_1^*, \ldots, y_{m_1}^*\}$).

Let $\big((\tilde{x}_2, \ldots, \tilde{x}_{m_1}), (\tilde{y}_1, \ldots, \tilde{y}_{m_2})\big)$ denote the output of $allocate(rev(P_2 \setminus \{x_1^*\}), rev(P_1 \setminus \{x_1^*\}), rev(o_2 \to \cdots \to o_m))$, where the order of the items are defined similar to the order in $\big(\{x_2, \ldots, x_{m_1}\}, \{y_1, \ldots, y_{m_2}\}\big)$. By induction hypothesis, $(\{\tilde{x}_2, \ldots, \tilde{x}_{m_1}\}, \{\tilde{y}_1, \ldots, \tilde{y}_{m_2}\})$ is the only SPNE allocation of the subgame where agent 1 chooses x_1^* (in the first stage of (P_1, P_2, O). It follows that $\{x_2^*, \ldots, x_{m_1}^*\} = \{\tilde{x}_2, \ldots, \tilde{x}_{m_1}\}$.

Now, suppose x_1^* is allocated in the k-th step of $allocate(rev(P_2), rev(P_1), rev(O))$. (Since $x_1^* \neq x_1$, we have that $k \neq m$.) We have the following two cases.

Case 1: $k \geq 2$. We note that the first $k-1$ items allocated in $allocate(rev(P_2), rev(P_1), rev(o_1 \to \cdots \to o_m))$ and in $allocate(rev(P_2 \setminus \{x_1^*\}), rev(P_1 \setminus \{x_1^*\}), rev(o_2 \to \cdots \to o_m))$ are the same. Let $\big((x_{k_1}, \ldots, x_{m_1}), (y_{k_2}, \ldots, y_{m_2})\big)$ denote these $k-1$ items allocated to agent 1 and 2 respectively. Let $G' = \{x_{k_1}, \ldots, x_{m_1}, y_{k_2}, \ldots, y_{m_2}\}$. We also note that $\big((\tilde{x}_2, \ldots, \tilde{x}_{k_1-1}), (\tilde{y}_1, \ldots, \tilde{y}_{k_2-1})\big)$ is the outcome of $allocate(rev(P_2 \setminus (G' \cup \{x_1^*\})), rev(P_1 \setminus (G' \cup \{x_1^*\})), rev(o_2 \to \cdots \to o_{m+1-k}))$. Hence, by the induction hypothesis $\big(\{\tilde{x}_2, \ldots, \tilde{x}_{k_1-1}\}, \{\tilde{y}_1, \ldots, \tilde{y}_{k_2-1}\}\big)$ is the unique SPNE allocation of the sub-game $(P_1 \setminus (G' \cup \{x_1^*\}), P_2 \setminus (G' \cup \{x_1^*\}), o_2 \to \cdots \to o_{m+1-k})$. By the induction hypothesis, $\big((x_1, \ldots, x_{k_1-1}), (y_1, \ldots, y_{k_2-1})\big)$, which is the output of $allocate(rev(P_2 \setminus G'), rev(P_1 \setminus G'), rev(o_1 \to \cdots \to o_{m+1-k}))$, is the unique SPNE allocation of the game $(P_1 \setminus G', P_2 \setminus G', o_1 \to \cdots \to o_{m+1-k})$.

Therefore, in the first stage of the game $(P_1 \setminus G', P_2 \setminus G', o_1 \to \cdots \to o_{m+1-k})$, if agent 1 chooses x_1^* rather than x_1, then she will not be strictly better off. We have that $u_1(x_1, \ldots, x_{k_1-1}) \geq u_1(x_1^*, \tilde{x}_2, \ldots, \tilde{x}_{k_1-1})$, which means that $u_1(\{x_1, \ldots, x_{m_1}\}) \geq u_1(x_1^*, \tilde{x}_2, \ldots, \tilde{x}_{m_1})$, and the equality holds if and only if $\{x_1, \ldots, x_{k_1-1}\} = \{x_1^*, \tilde{x}_2, \ldots, \tilde{x}_{k_1-1}\}$ (which is equivalent to $\{x_1, \ldots, x_{m_1}\} = \{x_1^*, \tilde{x}_2, \ldots, \tilde{x}_{m_1}\}$). This is a contradiction.

Case 2: $k = 1$. We have the following two subcases.

Case 2.1: $o_m = 1$. In this case, x_1^* is on the top of $rev(P_2)$, which means that it is the *least* preferred item of agent 2. Let b denote the second least-preferred item of agent 2. Let $\big((x_2' \ldots, x_{m_1}'), (y_1' \ldots, y_{m_2}')\big)$ denote the outcome of $allocate(rev(P_2 \setminus \{b\}), rev(P_1 \setminus \{b\}), rev(o_2 \cdots o_m))$. We next show that $\big((b, x_2' \ldots, x_{m_1}'), (y_1' \ldots, y_{m_2}')\big)$ is also an SPNE of (P_1, P_2, O). To this end we make the following observations on $allocate(rev(P_2 \setminus \{b\}), rev(P_1 \setminus \{b\}), rev(o_2 \cdots o_m))$ and $allocate(rev(P_2 \setminus \{x_1^*\}), rev(P_1 \setminus \{x_1^*\}), rev(o_2 \cdots o_m))$.

1. The first item allocated to agent 1 in $allocate(rev(P_2 \setminus \{x_1^*\}), rev(P_1 \setminus \{x_1^*\}), rev(o_2 \cdots o_m))$ is b. That is, $\tilde{x}_{m_1} = b$.
2. The first item allocated to agent 1 in $allocate(rev(P_2 \setminus \{b\}), rev(P_1 \setminus \{b\}), rev(o_2 \cdots o_m))$ is x_1^*. That is, $x_{m_1}' = x_1^*$.
3. In the remaining iterations, both mechanisms allocate the same item to the same agent.

It follows that $\{b, x_2' \ldots, x_{m_1}'\} = \{x_1^*, \tilde{x}_2, \ldots, \tilde{x}_{m_1}\}$, which means that in the first stage of the game (P_1, P_2, O), agent 1 obtain the same utility for choosing b and x_1^* (while the rest of the allocation is computed by backward induction). Therefore $\big((b, x_2' \ldots, x_{m_1}'), (y_1' \ldots, y_{m_2}')\big)$ is also an SPNE. Notice that b is *not* allocated in the first round of $allocate(rev(P_2), rev(P_1), rev(O))$. As we discussed in Case 1, if $\{b, x_2' \ldots, x_{m_1}'\} \neq \{x_1, \ldots, x_{m_1}\}$, then $\big((b, x_2' \ldots, x_{m_1}'), (y_1' \ldots, y_{m_2}')\big)$ cannot be an SPNE, which contradicts the assumption.

Case 2.2: $o_m = 2$. In this case, x_1^* is the *least* preferred item of agent 1. Intuitively it implies that $\big(\{x_1^*, \tilde{x}_2, \ldots, \tilde{x}_{k_1-1}\}, \{\tilde{y}_1, \ldots, \tilde{y}_{k_2-1}\}\big)$ is not an SPNE of (P_1, P_2, O), because in the first stage agent 1 picks her least preferred item. Formally, let c denote the second least-preferred item of agent 1. Let $\big((x_2' \ldots, x_{m_1}'), (y_1' \ldots, y_{m_2}')\big)$ denote the outcome of $allocate(rev(P_2 \setminus \{c\}), rev(P_1 \setminus \{c\}), rev(o_2 \cdots o_m))$. We make the following observations on $allocate(rev(P_2 \setminus \{c\}), rev(P_1 \setminus \{c\}), rev(o_2 \cdots o_m))$ and $allocate(rev(P_2 \setminus \{x_1^*\}), rev(P_1 \setminus \{x_1^*\}), rev(o_2 \cdots o_m))$.

1. The first item allocated to agent 2 in $allocate(rev(P_2 \setminus \{x_1^*\}), rev(P_1 \setminus \{x_1^*\}), rev(o_2 \cdots o_m))$ is c. That is, $\tilde{y}_{m_2} = c$.
2. The first item allocated to agent 2 in $allocate(rev(P_2 \setminus \{c\}), rev(P_1 \setminus \{c\}), rev(o_2 \cdots o_m))$ is x_1^*. That is, $y_{m_2}' = x_1^*$.
3. In the remaining iterations, both mechanisms allocate the same item to the same agent.

It follows from Point 3 above that for each $2 \leq i \leq m_1$, $\tilde{x}_i = x_i'$. Because $u_1(c) > u_1(x_1^*)$, $u_1(c, x_2' \ldots, x_{m_1}') > u_1(x_1^*, \tilde{x}_2 \ldots, \tilde{x}_{m_1})$, which contradicts the assumption that the latter is an SPNE. ■

Three agents

With three agents, there may no longer be an unique subgame perfect Nash equilibrium.

Example 3 *Suppose agent 1 and 3 have the preference ordering $1 > 2 > 3 > 4$, and agent 2 has $3 > 4 > 1 > 2$. We consider the policy 1231. There are two SPNE. In the first, all agentspick sincerely. Agent 1 gets items 1 and 4, agent 2 gets item 3 and agent 3 gets item 2. In the other SPNE, agent 1 first picks item 3 strategically. Agent 2 and 3 (who have no incentive but to pick sincerely) get items 4 and 1 respectively, leaving item 2 for agent 1. With Borda utilities, agent 1 has the same utility in both equilibria.*

This example can be adapted to demonstrate that the SPNE depend on the actual utilities, and not (as in the two agent case) just on the preference ordering.

Example 4 *Consider the same example as before. Suppose we have a super-linear utility function that assigns $(m-i)^2$ utility for the ith ranked item out of m. In this case, there is an unique SPNE in which agent 1 picks sincerely and gets items 1 and 4. On the other hand, suppose we have a sub-linear utility function that assigns $\sqrt{m-i}$ utility for the ith ranked item out of m. In this case, there is an unique SPNE in which agent 1 picks strategically and gets items 3 and 2.*

Thus, any mechanism for computing the subgame perfect Nash equilibria for 3 or more agents must take into account the actual utilities assigned to items by agents and cannot use just the preference orderings. We can also glue many copies of this example together to show that there can be an exponential number of subgame perfect Nash equilibria.

Theorem 2 *With 3 agents and $4m$ items, there exists an allocation problem with Borda utilities and 2^m subgame perfect Nash equilibria.*

Proof: Suppose agent 1 and 3 have the preference ordering $1 > 2 > 3 > 4$, and agent 2 has the ordering $3 > 4 > 1 > 2$. We consider the policy 1231. Then there are, as argued previously, two SPNE depending on whether agent 1 picks first item 1 sincerely or item 3 strategically. We now repeat this construction for items 5 to 8 with the rotation of agents that maps agent 1 onto 2, 2 onto 3 and 3 onto 1. That is, we extend the policy with 2312 (the rotation of 1231) to give the combined policy 12312312. The preference ordering of agent 1 is now $1 > 2 > 3 > 4 > 5 > 6 > 7 > 8$, of agent 2 is $3 > 4 > 2 > 1 > 5 > 6 > 7 > 8$, and of agent 3 is $1 > 2 > 3 > 4 > 7 > 8 > 1 > 2$. This introduces new SPNE in which agent 2 either chooses item 5 sincerely or item 7 strategically on their second pick. As this choice is independent of agent 1's first sincere or strategic choice of the items 1 or 3, we now have 4 possible SPNE. We can repeat this extension of the allocation problem with 4 more items, rotating the roles of the agents as before. Each such extension doubles the number of SPNE. ■

An interesting open question is whether we can efficiently compute a subgame perfect Nash equilibrium with three (or a small number of) agents. We will shortly identify a case where we can efficiently compute (one or all of) the subgame perfect Nash equilibria with three or more agents.

Many agents

As argued previously, we can use backward induction to compute the subgame perfect Nash equilibria. However, the game tree can be prohibitively large. For instance, with $2m$ items to be divided between m agents, the game tree contains m^m branches. Brams and Straffin (1979) argue that "*no algorithm is known which will produce optimal play more efficiently than by checking many branches of the game tree*". We prove here that computing the subgame perfect Nash equilibrium is in fact PSPACE-hard.

Theorem 3 *With an unbounded number of agents and Borda utilities, computing the subgame perfect Nash equilibrium is PSPACE-hard.*

Proof: Backward induction shows that it is in PSPACE. To show hardness, we give a reduction from QSAT. We are given a quantified formula $\exists x_1 \forall x_2 \exists x_3 \cdots \forall x_q \,.\, \varphi$ where q is even and ask whether the formula is true. Let $\varphi = C^1 \wedge \ldots \wedge C^t$, where C^j is a 3 clause, $l_j^1 \vee l_j^2 \vee l_j^3$. We construct a sequential allocation instance in which there is an unique SPNE with a utility to the first player larger than a threshold if and only if the formula is true. In the sequential allocation instance, there are q agents who represent the binary variables. Each of these agents choosing one out of two items represents a valuation of the variable. The agents that correspond to $\exists$ quantifiers (that is, agents 1, 3,. . ., $q-1$) obtain higher utility if φ is true under the current valuation, and the agents that correspond to $\forall$ quantifiers (that is, agents 2, 4,. . ., q) obtain higher utility if φ is false under the current valuation. There are also some other agents that are used to encode the QSAT instance, which we will specify later.

Let a be an item, and k, p be natural numbers. We define an ordering $O_p^k(a)$ as follows. It introduces $2k+1$ new agents $A_p^1, \ldots, A_p^{2k+1}$ and $5k+1$ new items $\{a_p, b_p^1, \ldots, b_p^k, c_p^1, \ldots, c_p^k, d_p^1, \ldots, d_p^k, e_p^1, \ldots, e_p^k, f_p^1, \ldots, f_p^k\}$. The preferences of the new agents are as follows:

Agent	Preferences
A_p^1	$b_p^1 > c_p^1 > d_p^1 > e_p^1 >$ Others
$\vdots$	$\vdots$
A_p^k	$b_p^k > c_p^k > d_p^k > e_p^k >$ Others
A_p^{k+1}	$c_p^1 > f_p^1 >$ Others
$\vdots$	$\vdots$
A_p^{2k}	$c_p^k > f_p^k >$ Others
A_p^{2k+1}	$a > b_p^k > \cdots > b_p^1 > a_p >$ Others

In $O_q^k(a)$, a is the item that we want to "duplicate", k is the number of duplicates, and q is merely an index. We can prove by induction that if a has not chosen (in previous rounds), then after agents have chosen items according to such an ordering, $\{f_p^1, \ldots, f_p^k\}$ will be chosen and $\{d_p^1, \ldots, d_p^k\}$ will not be chosen ; if a has been chosen, then $\{d_p^1, \ldots, d_p^k\}$ will be chosen rather than $\{f_p^1, \ldots, f_p^k\}$.

We now specify the sequential allocation instance by using the orderings $O_p^k(a)$. All agents introduced in $O_p^k(a)$ will not appear in other places in the ordering. For each $i \leq q$, there are two items 0_i and 1_i that represent the values of x_i, and there is an agent A_i corresponding to the valuation and another agent B_i that is used to make sure that A_i will choose either 0_i or 1_i in the ith round. For each $i \leq q$, D_i is an agent whose preferences are $d_i >$ Others, where d_i is a new item that is used to create a "gap" between items available to agent A_i. The first $(2t+4)q$ agents are the following: $D_1 > \cdots > D_q > A_1 > \cdots > A_q > O_1^t(0_1) > \cdots > O_q^t(0_q) > B_1 > \cdots > B_q$. The preferences of B_i are $0_i > 1_i >$ Others. The preferences of A_i will be defined after we specify the entire ordering and define all items. For notational convenience, for each $i \leq q$ and each $j \leq t$ we rename d_i^j to be 0_i^j, and rename f_i^j to be 1_i^j.

For each clause C^i, we have an agent C_i. Suppose v_{j_1}, v_{j_2}, and v_{j_3} correspond to the three valuations that makes C_i true, then we let the preferences of C_i to be $v_{j_1}^i > v_{j_2}^i > v_{j_3}^i > g > g_i' >$ Others, where g and g_i' are new items. g is used to detect whether any clause is not satisfied. For example, suppose $C^i = x_1 \vee \neg x_2 \vee x_3$, then the preferences of C_i are $1_1^i > 0_2^i > 1_3^i > g > g_i' >$ Others. The remaining agents in the ordering are:

$$C_1 > \cdots > C_t > O_{q+1}^q(g) > A_1 > \cdots > A_q$$

The agents and new items introduced in $O_{q+1}^q(g)$ are used to impose "feedback" to A_1 through A_q, such that if g is allocated before $O_{q+1}^q(g)$ (which means that the formula is not satisfied under the valuation encoded in the first q rounds), then some items that are more valuable to the agents that correspond to the $\forall$ quantifier are made available; if g is not allocated before $O_{q+1}^q(g)$, then some items that are more valuable to the agents that correspond to the $\exists$ quantifier are made available. Finally, for each $i \leq q$, we define A_i as follows.

- If i is odd, then $A_i = 0_i > 1_i > d_{q+1}^i > d_i > f_{q+1}^i$;
- if i is even, then $A_i = 0_i > 1_i > f_{q+1}^i > d_i > d_{q+1}^i$.

To summarise, in the sequential allocation instance, there are $3q + t + (2t+1)q + 2q + 1$ agents and $m = 3q + (5t+1)q + 1 + t + 5q + 1$ items, which are polynomial in the size of the formula ($\Omega(t+q)$). The items are summarised in the following table.

for	Item	Introduced in
$i \leq q$	d_i	D_i
$i \leq q$	$0_i, 1_i$	A_i
$i \leq q, j \leq t$	a_i b_i^j c_i^j d_i^j (a.k.a. 0_i^j) e_i^j f_i^j (a.k.a. 1_i^j)	$O_i^j(0_i)$
	g	C_1
$j \leq t$	g_t'	C_j
$j \leq q$	$a_{q+1}, b_{q+1}^j, c_{q+1}^j,$ $d_{q+1}^j, e_{q+1}^j, f_{q+1}^j$	$O_{q+1}^q(g)$

The final ordering over agents is

$$D_1 > \cdots > D_q > A_1 > \cdots > A_q > O_1^t(0_1) > \cdots > O_q^t(0_q)$$
$$> B_1 > \cdots > B_q > C_1 > \cdots > C_t > O_{q+1}^q(g)$$
$$> A_1 > \cdots > A_q$$

If we must allocate all items then we can add some dummy agents to the end of the ordering.

If an agent only appears once in the ordering, then it is her strictly dominant strategy to pick her most preferred available item. In any SPNE, in the first q rounds $d_1, \ldots, d_q$ will be chosen. In the next q rounds, agent i must choose either 0_i or 1_i, otherwise 0_i will be chosen by agent A_i^{2t+1} introduced in $O_i^t(0_i)$ and 1_i will be chosen by B_i. Hence, the choices of agents A_i correspond to valuations of the variables, and these valuations are duplicated by $O_i^t(0_i)$ that will be used to satisfy clauses. (We note that if A_i chooses 0_i, then after $O_i^t(0_i)$, $\{0_i^1, \ldots, 0_i^t\}$ are still available, but $\{1_i^1, \ldots, 1_i^t\}$ are not available; and vice versa.) Then, a clause C^i is satisfied if and only if at least one of the top 3 items of agent C_i is available (otherwise C_i chooses g). Hence, after agent C_t, g is available if and only if all clauses are satisfied. Finally, if g is available after agent C_t, then the agents that correspond to the $\exists$ quantifiers can choose d_{q+1}'s to increase their utility by 3, but the agents that correspond to the $\forall$ quantifiers can only choose d_{q+1}'s to increase their utility by 1; and vice versa. Hence, the agents that correspond to $\exists$ quantifiers will choose valuations to make F true, while the agents that correspond to $\forall$ quantifiers will choose valuations to make F false. It is easy to see that there is an unique SPNE in which agent 1's utility is at least $2m-5$ (that is, she gets one of $\{0_1, 1_1\}$ and d_{q+1}^1) if and only if the formula is true. ■

Similar utilities

When agents have the same utilities for all items, there is no incentive for them to act other than sincerely (Proposition 6 in (Bouveret and Lang 2011)). Strategic behaviour is only worthwhile when agents have different utilities. For example, if you value an item that I don't, you may strategically delay choosing it since it might still be available in a later round. We say that an item is *multi-valued* if two agents assign it different utilities. Otherwise we say that the item is *single valued*. Suppose there are only a small number of multi-valued items. Then once the multi-valued items are allocated, agents have no incentive to act strategically and we can compute the subgame perfect Nash equilibrium easily by sincerely allocated the remaining single valued items. We construct an algorithm for computing subgame perfect Nash equilibria which exploits this fact. More precisely, we exploit the following result.

Theorem 4 *In any subgame perfect Nash equilibrium, we can permute the single valued items so that every agent either picks a multi-valued item or the single valued item that remains with the greatest utility.*

Proof: Suppose there is a SPNE in which an agent picks an item that does not have the greatest utility amongst the remaining single valued items. If another agent picks this item, then when we permute these two items, the allocation is strictly more preferred. Hence, the original allocation was not a SPNE since there was a better move available. The agent currently picking an item must also have picked the remaining single valued item with the greatest utility at a later choice point. When we permute these two items, the allocation has the same utility for this agent. Hence, it is also a SPNE. ■

Proposition 6 in (Bouveret and Lang 2011) follows quickly from this result. By exploiting this result, we can build an efficient algorithm for computing the subgame perfect Nash equilibria for any number of agents when the number of multi-valued items is small.

Theorem 5 *For any number of agents, we can compute a subgame perfect Nash equilibrium in $O(k!m^{k+1})$ time where k is the number of multi-valued items, and m is the number of single valued items.*

Proof: Let $T(k,m)$ be the number of steps to compute a SPNE. Suppose there are no multi-valued items. Then we can compute the unique SPNE in m steps by simply computing the sincere allocation. Hence $T(0,m) = m$.

Suppose there are no single valued items. Then we can compute a SPNE in at most $k!$ steps by considering all possible elimination orders for the items. Hence $T(k,0) = O(k!)$.

Suppose there is just one multi-valued item. Consider using backward induction to compute a SPNE. At the first step, the agent picking chooses either the one multi-valued item or the single valued item of greatest utility. In the first case, we now have a subproblem with no multi-valued items and we can compute the unique SPNE in m steps. In the second case, we again have a subproblem with just one multi-valued item but now with $m-1$ single valued items. That is, $T(1,m) = T(0,m) + T(1,m-1) = m + T(1,m-1)$. We compare the utility of the two allocations that result from either case, and keep as a SPNE whichever has the larger utility for the agent picking. If the two allocations have the same utility, we choose one arbitrarily.

More generally, suppose there are k multi-valued items where $k \geq 1$. By a similar argument to the case of one multi-valued item, we have $T(k,m) = k * T(k-1,m) + T(k,m-1)$. Solving these recurrences gives $T(k,m) = O(k!m^{k+1})$. ■

If k is small, if follows that we can efficiently compute a subgame perfect Nash equilibrium. Note that we can adapt this method to compute *all* subgame perfect Nash equilibria by collecting together allocations with the same maximal utility for the agent who is picking.

Externalities

In some situations, you may care about both the items you receive and the items that the other agents receive. For example, if we are using this sequential allocation procedure to pick two football teams, one agent wants an allocation that gives them the best possible team *and* gives the other agent the worst possible team. Kohler and Chandrasekaran (1971) consider a simple setting with two agents, the strict alternating policy, and the payoff to a player is the sum of utilities of items that this player receives less the sum of utilities of items that the other player receives. This is a zero-sum sequential game. They prove by induction that the optimal strategy for both players is simply to pick the item with the maximum sum of utilities for both players. We now argue that this result generalizes to any policy (and not just the simple alternating policy).

Theorem 6 *With two agents, any policy and such a payoff, the optimal strategy for both agents is to pick the item with the maximum sum of utilities.*

Proof: Suppose agent 1 selects the items A out of the set B, then their payoff is $\sum_{i\in A} u_1(i) - \sum_{i\in B-A} u_2(i)$ where $u_j(i)$ is the utility of item i for agent j. This equals $\sum_{i\in A} u_1(i) - (\sum_{i\in B} u_2(i) - \sum_{i\in A} u_2(i))$. This simplifies to $\sum_{i\in A}(u_1(i) + u_2(i)) - c$ where c is a constant. Hence the objective of agent 1 is to maximise this objective. Similarly, agent 2 has the same objective. To simplify the proof, we suppose that if an agent picks two or more items successively then the agent picks the items in order of their sum of utilities. The proof uses induction on the length of the policy. Trivially, the strategy of picking the item with maximum sum of utilities is optimal for two of fewer items. Suppose the strategy is optimal for $n-1$ items and consider a game of n items. WLOG we suppose agent 1 picks first in this game (otherwise we can simply reorder the agents). There are two cases. In the first case, agent 1 picks the item with maximum sum of utilities. By the induction hypothesis, we are done. In the second case, agent 1 does not pick the item with maximum sum of utilities. By the induction hypothesis and the assumption that successive picks by the same agent are of items in decreasing order of the sum of their utilities, the item with maximum sum of utilities is picked by agent 2. Indeed, the items picked by agent 2 now either have a better or equal sum of utilities as in the first case. Hence agent 2 does better. As this is a zero-sum game, agent 1 therefore does worse. Hence, this is not the optimal strategy. ■

With externalities, everyone picks the items which are most highly rated overall (or equivalently, with the best average rating). It is easy to see why sincere behaviour is optimal in this case. Both agents share the same objective, and strategic behaviour requires different preference orderings.

Strategy proof mechanisms

In (Brams and Kaplan 2004), Brams and Kaplan re-design the sequential allocation mechanism in which two agents strictly alternate picking items to make it strategy proof. The modification is as follows. Whenever an agent picks an item, they can offer to swap it for an item already selected by the other agent. This offer to swap items remains until all items are allocated. At this point, the offers are considered in reverse order, from the most recent to the most ancient. Informally this works as follows. Suppose an agent picks an item strategically out of order, expecting the second agent not to pick the first agent's most preferred item. Then the second can immediately pick the first agent's most preferred item and offer to swap it for the item just picked. At the end of the game, the first agent will accept such an offer as it improves their outcome. In this way, the second agent negates the first agent's strategic move. We now show that this modification gives a mechanism that is strategy proof whatever policy is used (and not use the strict alternating policy).[1]

Theorem 7 *For any policy involving two agents, the subgame perfect Nash equilibrium for this re-designed sequential allocation procedure is the sincere allocation*

Proof: To simplify the proof, we suppose that when agents make successive picks, they do so in preference order. The proof uses induction on the number of items. Clearly it holds for one item. Suppose it holds for $n-1$ items and consider dividing n items. WLOG we suppose agent 1 picks first. There are two cases. In the first, agent 1 picks their most preferred item. By the induction hypothesis, the step case holds. In the second, agent 1 picks an item, call it a, that is not their most preferred (which we call b). Considering the remaining $n-1$ items. By the induction hypothesis, the SPNE is the sincere allocation for these items. There are two cases. In the first case, agent 2 is allocated b in this outcome. Now agent 1 will gladly swap a for b if agent 2 offers it, resulting in the sincere allocation for all n items. Suppose agent 2 prefers b to a then this swap will not be offered. This is a worse outcome for agent 1 so cannot be the SPNE. Hence, in this subcase, agent 1 will not make this strategic move of picking a in the first pace. In the second case, agent 1 is allocated b in this outcome. By the induction hypothesis, the remaining $n-1$ items are allocated sincerely. The resulting allocation of n items is equivalent to one in which agent 1 picked truthfully b and then a. ■

It remains an interesting open question how to make the sequential allocation procedure strategy proof for three (or more) agents.

Conclusions

We have studied a simple sequential allocation procedure for allocating indivisible goods. We have proved for two agents and additive utilities, computing the unique subgame perfect Nash equilibrium takes linear time irrespective of the policy. For more agents, even with Borda utilities, we argued that there can be an exponential number of subgame perfect Nash equilibria, and computing even one of them is PSPACE-hard. A special case is when agents value most of the items identically. In this case, we give an efficient procedure for computing one (or all) of the subgame perfect Nash equilibria. We have also considered strategic behavior when agents have externalities (and value items assigned to the other agents), and modifications to the mechanism to make it strategy proof.

Acknowledgments

Lirong Xia is supported by NSF under Grant 1136996 to the Computing Research Association for the CIFellows Project. Nina Narodytska and Toby Walsh are supported by the Australian Governments Department of Broadband, Communications and the Digital Economy, the Australian Research Council and the Asian Ofce of Aerospace Research and Development through grants AOARD-124056.

[1] Brams and Kaplan suggest that their modification of the sequential allocation procedure works for policies other than strict alternating. However, their proof in the Appendix of (Brams and Kaplan 2004) supposes just strict alternation.

References

Bouveret, S., and Lang, J. 2011. A general elicitation-free protocol for allocating indivisible goods. In Walsh, T., ed., *Proc. of 22nd International Joint Conference on Artificial Intelligence (IJCAI 2011)*. IJCAI/AAAI.

Brams, S., and Fishburn, P. 2000. Fair division of indivisible items between two people with identical preferences: Envy-freeness, Pareto-optimality, and equity. *Social Choice and Welfare* 17(2):247–267.

Brams, S., and Kaplan, T. 2004. Dividing the indivisible. *Journal of Theoretical Politics* 16(2):143–173.

Brams, S.; Edelman, P.; and Fishburn, P. 2003. Fair division of indivisible items. *Theory and Decision* 55(2):147–180.

Brams, S., and Straffin, P. 1979. Prisoners' dilemma and professional sports drafts. *The American Mathematical Monthly* 86(2):pp. 80–88.

Budish, E., and Cantillon, E. 2007. Strategic behavior in multi-unit assignment problems: Theory and evidence from course allocations. In Cramton, P.; Müller, R.; Tardos, E.; and Tennenholtz, M., eds., *Computational Social Systems and the Internet*, number 07271 in Dagstuhl Seminar Proceedings. Internationales Begegnungs- und Forschungszentrum für Informatik (IBFI), Schloss Dagstuhl, Germany.

Budish, E.; Che, Y.-K.; Kojima, F.; and Milgrom, P. 2013. Designing random allocation mechanisms: Theory and applications. *American Economic Review*. Forthcoming.

Kohler, D., and Chandrasekaran, R. 1971. A Class of Sequential Games. *Operations Research* 19:270–277.

Othman, A.; Sandholm, T.; and Budish, E. 2010. Finding approximate competitive equilibria: efficient and fair course allocation. In van der Hoek, W.; Kaminka, G.; Lespérance, Y.; Luck, M.; and Sen, S., eds., *9th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2010)*, 873–880. IFAAMAS.