Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence (IJCAI 2015)

A Dictatorship Theorem for Cake Cutting*

Simina Brânzei and Peter Bro Miltersen
Department of Computer Science
Aarhus University, Denmark

Abstract

We consider discrete protocols for the classical Steinhaus cake cutting problem. Under mild technical conditions, we show that any deterministic strategy-proof protocol for two agents in the standard Robertson-Webb query model is dictatorial, that is, there is a fixed agent to which the protocol allocates the entire cake. For $n > 2$ agents, a similar impossibility holds, namely there always exists an agent that gets the empty piece (i.e. no cake). In contrast, we exhibit randomized protocols that are truthful in expectation and compute approximately fair allocations.

1 Introduction

In this paper, we consider the classical *cake cutting* problem due to Steinhaus [1948] :

> *How can one fairly allocate a divisible good among multiple agents with private heterogeneous valuations?*

Cake cutting is a fundamental problem in fair division. The *cake* is a metaphor for a heterogeneous divisible resource, such as land, time, memory in shared computing systems, clean water, greenhouse gass emissions, fossil fuels and other natural deposits. The problem is to fairly divide the resource among multiple participants, such that everyone is happy with their allocation.

The model has been studied in a large body of literature in economics, political science, and mathematics [Brams and Taylor, 1996; Robertson and Webb, 1998], and has recently been studied in the computer science community, as problems in resource allocation and fair division in particular are arguably relevant for the design of multiagent systems. Examples include manufacturing and scheduling, airport traffic, and industrial procurement [Chevaleyre *et al.*, 2006; Procaccia, 2013]. More recently, the problem of fair division is also motivated by the allocation of computational resources (such as CPU, memory, bandwidth) among users of shared computing systems [Gutman and Nisan, 2012; Kash *et al.*, 2013], and has emerged as an important topic in artificial intelligence [Procaccia, 2009; Caragiannis *et al.*, 2011; Cohler *et al.*, 2011; Brams *et al.*, 2012; Bei *et al.*, 2012; Aumann *et al.*, 2013; Kurokawa *et al.*, 2013; Brânzei *et al.*, 2013; Chen *et al.*, 2013].

Mathematically, the cake is modeled as the interval $[0, 1]$ and must be divided among a set $N = \{1, \ldots, n\}$ of agents. A *piece of cake* X is a finite set of disjoint subintervals of $[0, 1]$. The preferences of each agent i are given by an integrable, non-negative *value density* function $v_i : [0, 1] \to \mathbb{R}^+$ that induces a *value* for each possible piece of cake. The valuation of agent i for a piece X is given by $V_i(X) = \int_X v_i(x)dx$. Without loss of generality, we assume that each agent $i \in N$ has a value of one for the entire cake: $V_i([0, 1]) = 1$.

An *allocation* $A = (A_1, \ldots, A_n)$ is a partition of the cake among the agents, that is, each agent i receives the piece A_i, the pieces are disjoint[1], and $\bigcup_{i \in N} A_i = [0, 1]$. The classical literature on cake cutting is concerned with obtaining *fair* allocations; among the many existing criteria of fairness we mention proportional, envy-free, and perfect partitions. For instance, an allocation A is said to be *envy-free* if for all agents $i, j \in N$, $V_i(A_i) \geq V_i(A_j)$. In the classical literature on cake cutting, a protocol is said to have a property such as envy-freeness if each agent i is *guaranteed* not to be envious by behaving *truthfully* in the protocol (i.e., not misrepresenting its private valuation function), regardless of what the other agents do.

The classical cake cutting protocols, extensively discussed by Robertson and Webb [1998], can be divided in two main classes, namely *discrete* and *moving-knife* (or continuous) protocols. More recently, also *direct revelation* protocols were studied [Chen *et al.*, 2013; Mossel and Tamuz, 2010; Maya and Nisan, 2012]. Discrete protocols enjoy a standard query model, *the Robertson-Webb model*, that captures all existing discrete protocols and models the interaction be-

*Work supported by the Danish National Research Foundation and The National Science Foundation of China (under the grant 61061130540) through the Sino-Danish Center for the Theory of Interactive Computation (CTIC) and by the Danish Strategic Research Council through the Center for Research in Foundations of Electronic Markets (CFEM).

[1]For convenience, we actually allow pieces to intersect in a finite number of points; note that by definition, single points have zero value to all the agents.

tween the center and the agents through a sequence of queries. The Robertson-Webb model was employed in a body of work studying the complexity of cake cutting [Edmonds and Pruhs, 2006b; 2006a; Woeginger and Sgall, 2007; Procaccia, 2009; Kurokawa *et al.*, 2013] and is also the focus of the present paper (see formal definition below).

The classical discrete protocols are not *strategy-proof* [Chen *et al.*, 2013; Brânzei and Miltersen, 2013; Kurokawa *et al.*, 2013], i.e., there are scenarios (possible behaviors of the other agents) in which it is possible for an agent to get a piece of strictly larger value by misrepresenting its valuation function than by behaving truthfully. This begs the question of whether alternative strategy-proof protocols can be constructed. Addressing this question, Kurokawa *et al* showed a negative result: For any number of agents $n \geq 2$, there is no Robertson-Webb protocol of complexity bounded only by a function of n (i.e., independent of the valuations) that is strategy-proof and computes an envy-free allocation.

The main results of our paper are impossibility theorems closely releated to the result of Kurokawa *et al.*, but rather than stating that no fair allocation can computed, we essentially state that no reasonable allocation can be computed *at all*; thus, the "unfairness conclusions" of our theorems are stronger. Also, we do not need to make any assumption about the complexity of the protocols.

For two agents, our result is particularly strong, with a conclusion similar to the classical *dictatorship* results of social choice theory, in particular the Gibbard-Satterthwaite theorem [Gibbard, 1973; Satterthwaite, 1975], which is a cornerstone of social choice theory and mechanism design.

Theorem 1. *Suppose a deterministic cake cutting protocol for two agents in the Robertson-Webb model is strategy-proof. Then, restricted to hungry agents, the protocol is a dictatorship.*

Theorem 2. *Suppose a deterministic cake cutting protocol for $n \geq 3$ hungry agents in the Robertson-Webb model is strategy-proof. Then, in every outcome associated with truthful reports by hungry agents, there is at least one agent that gets the empty piece (i.e., no cake).*

Here, we say that an agent i is *hungry* if its value density function v_i is hungry, i.e., satisfies $v_i(x) > 0$ for all x. We say that a protocol is a *dictatorship* if there is a fixed agent (the dictator) to whom the entire cake is allocated in all truthful executions of the protocol, no matter what the value density functions are[2]. We say that a protocol is *strategy-proof* if for every profile of value density function it holds that truthful reporting is a dominant strategy for each agent i, when the protocol is viewed as a complete and perfect information extensive form game where agents choose strategically what to report.

[2]This is consistent with the standard meaning of "dictatorship" in social choice theory: For all preference profiles, the social choice is the most preferred alternative of the dictator.

1.1 Comments on the Impossibility Theorems

The theorems refer to the Robertson-Webb model as formalized originally by Woeginger and Sgall [2007]. An alternative and slightly more permissive formalization is given by Procaccia [2013]. But in his formalization, protocols such as the following are allowed:

"*Allocate [0, 0.5) to agent 1 and [0.5,1] to agent 2*"

This protocol is clearly strategy-proof but not a dictatorship. The only difference between the two formalizations is that the Woeginger-Sgall version requires all cut points to be defined by the agents rather than by the center. This property is essential for the theorems and their proofs.

Strategy-proofness is a notion more commonly used for direct revelation protocols than for indirect revelation protocols like the ones consider here. The definition we use is our interpretation of the definition stated by Kurokawa *et al.*. A weaker (i.e., more permissive) notion would be to call an indirect revelation protocol strategy-proof if the corresponding (by the revelation principle) direct revelation protocol is strategy-proof, i.e., has truth telling being a dominant strategy. It is easy to see that this weaker notion is equivalent to truth telling being a *Nash equilibrium* of each complete information game defined by the protocol (see, e.g., Proposition 9.23 of Nisan [2007]). It is a very interesting open problem to extend our results to the weaker notion. One can observe that the negative result of Kurokawa *et al.* does extend.

The theorems trivially fail without the restriction to hungry agents. For instance, the following protocol can be formalized in the Robertson-Webb model and is strategy-proof but not a dictatorship (unless restricted to hungry agents, in which case agent 1 becomes the dictator):

> *Ask agent 1 for an initial segment of the cake worth as much as the entire cake to him. Assign that segment to agent 1 and the rest (if any) to agent 2.*

The conclusion of Theorem 2 cannot be improved to the protocol being a dictatorship. Indeed, consider the following protocol:

> "*Agent* 1 *cuts the cake in two pieces of equal value. Agent* 2 *takes the piece it prefers. Agent* 3 *takes the remaining piece.*"

Agent 1 never receives anything, so it has no incentive to misreport. Agent 2 can always select its most preferrred piece, so it has no incentive to lie either. Finally, agent 3 takes the remaining piece without making any report; thus the protocol is strategyproof. However, it is not a dictatorship.

1.2 Organization of the paper

The paper proceeds as follows: In Section 2 we define the Robertson-Webb model. In Section 3, we prove our two main theorems. In Section 4, we show that by considering randomized instead of deterministic protocols, there exist meaningful protocols in the Robertson-Webb model that are non-dictatorial and truthful-in-expectation.

$x \leftarrow Cut\left(1; \frac{1}{2}\right)$
$\alpha \leftarrow Eval\,(2; x)$
if $\left(\alpha \geq \frac{1}{2}\right)$ **then**
  allocate $[0, x]$ to Agent 1 and $[x, 1]$ to Agent 2
else
  allocate $[0, x]$ to Agent 2 and $[x, 1]$ to Agent 1
end if

Algorithm 1: Cut-and-Choose protocol

2 The Robertson-Webb model

The Robertson-Webb model, as formalized by Woeginger and Sgall [2007], allows the following two query types between the protocol and the agents:

- Cut($i;\alpha$): Agent i cuts the cake at a point y where $V_i([0, y]) = \alpha$. The point y becomes a *cut point*.
- Eval($i; y$): Agent i returns $V_i([0, y])$ where y is a previously made cut point.

The queries made by the protocol may depend on the outputs of previous queries. At termination, the cut points define a partition of the cake into a finite set of intervals that the protocol allocates to the agents in some specified way.

We illustrate the Robertson-Webb model with the well known *Cut and Choose* protocol, which computes an envy-free and proportional allocation for two agents: (i) Agent 1 cuts the cake in two pieces that it values equally, (ii) Agent 2 chooses its favorite piece, and (iii) Agent 1 takes the remainder. *Cut and Choose* can be defined in the Robertson-Webb model as shown in Algorithm 1.

To make the definition rigorous, we formally define a Robertson-Webb protocol as an *infinite decision tree* (see Figure 1) where each internal node $\mathcal{X}$ is labeled with the query made if $\mathcal{X}$ is reached. There is a directed outgoing edge e from such a node $\mathcal{X}$ for every possible answer to the given query (i.e., infinitely many), and the node $\mathcal{Y}$ reached through edge e is either an internal node, or a leaf containing the resulting allocation if the path to $\mathcal{Y}$ is taken. We require that the protocol does not ask for information it already knows and does not accept information from the agents that is inconsistent with previous replies (e.g., reports that implies negatively valued subintervals). We require that the protocol terminates (reaches a leaf) for every profile of value density functions, if agents report truthfully. If truthful reporting according to a value density function profile $\mathbf{v}$ makes the protocol reach a leaf u, we say that $\mathbf{v}$ *is associated with* u and vice versa.

Without loss of generality, protocols can be assumed to have alternating Cut and Eval queries: For any protocol $\mathcal{M}$, there is an equivalent protocol $\mathcal{M}'$ that after every cut asks for the values for all agents of the newly generated subintervals and produces allocations identical to $\mathcal{M}$ on every instance. We say that a protocol is *strategy-proof* if for every profile of value density function it holds that truthful reporting is a dominant strategy for each agent i, when the protocol is viewed as a complete and perfect information extensive form game where agents choose strategically what to report.

Viewing the protocol as a complete and perfect information extensive form game this way in particular entails assuming that all communication between agents and center is broadcast and accessible to all agents. In addition, the transformation of a protocol $\mathcal{M}$ to a protocol $\mathcal{M}'$ with alternating Cut and Eval queries as described above preserves strategy-proofness: any extra Eval queries introduced in the transformation are payoff irrelevant "cheap talk" seen from the point of view of the agents.

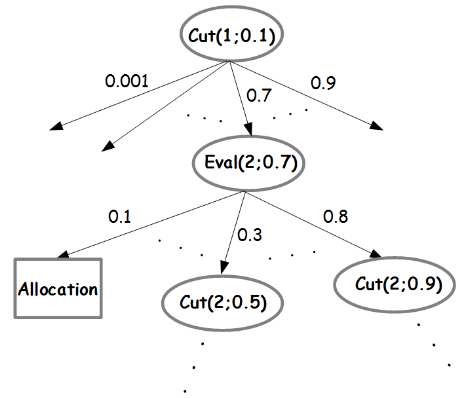

Figure 1: Representation of a Robertson-Webb protocol as an infinite decision tree

3 Proof of the main theorems

We start with a lemma.

Lemma 1. *Let $\mathcal{M}$ be a strategy-proof Robertson-Webb protocol for two agents that is not dictatorial when restricted to hungry valuations. Then, in no leaf of $\mathcal{M}$ reached under truthful reporting for some profile of hungry valuations, is the entire cake given to a single agent.*

Proof. Assume to the contrary that at a reachable leaf u the entire cake is allocated to a single agent, say, agent 1. Let $\mathbf{v} = (v_1, v_2)$ be the profile of hungry value density functions associated with u. Since the protocol is not dictatorial when restricted to hungry agents, there is another reachable leaf u' where agent 1 does not receive the entire cake. Let $\mathbf{v}' = (v_1', v_n')$ be a profile of hungry value density functions associated with u'. Consider now the outcome when agents report according to the profile $w = (v_1', v_2)$. It must be the case that agent 1 receives the entire cake in this outcome; otherwise the protocol is not strategy-proof, as agent 1 could misrepresent his value density function as v_1 and get the entire cake, assuming that the other agent reports according to v_2. But this means that when the protocol is played with profile w, agent 2 would benefit from misrepresenting his value density function as v_2' rather than v_2, as he would then receive a non-empty piece rather than nothing at all. This contradicts the strategy-proofness of $\mathcal{M}$. □

For the proof of the theorems, it is convenient to define a restricted kind of protocols where the physical locations of the cut points do not matter; instead, the protocol is only concerned with the values that the agents have for the generated pieces. We call such protocols *strictly mediated* and observe that in fact, all classical protocols in the Robertson-Webb model belong to this class. Strict mediation can be interpreted as the center not having direct access to the cake; instead, it can only see it through the eyes of the agents.

Definition 1 (*Strictly Mediated Protocol*). *A* strictly mediated protocol *for n agents is an infinite decision tree containing two kinds of (internal) nodes –* Cut *nodes and* Eval *nodes – and leaves:*

- *An* Eval *node is labeled by a pair of natural numbers (i, j) and has a successor for each real number $\alpha \in (0, 1)$.*
- *A* Cut *node is labeled by a pair of natural numbers (i, j) and a real number $\alpha \in (0, 1)$ and has a* single *successor.*
- *Each leaf is labeled with a finite sequence of natural numbers in $\{1, \dots, n\}$.*

The semantics is the following. At any point in the execution (at some node u in the tree), a set of cut points $x_0 = 0 < x_1 < x_2 < \cdots < x_k < x_{k+1} = 1$ has been defined (where k is the number of cut nodes above u):

- *When an* Eval *node X with labels (i, j) is reached, agent i is asked for its value of interval $[x_j, x_{j+1}]$; given the agent's answer, $\alpha \in (0, 1)$, execution moves to the successor node reached along the edge labeled with the value α.*
- *When a* Cut *node with labels (i, j, α) is reached, agent i is asked to define a new cut point x' somewhere between x_j and x_{j+1} so that his value of the interval $[x_j, x']$ is an α-fraction of his value of the interval $[x_j, x_{j+1}]$.*
- *When a leaf node with labels $(i_0, i_1, \dots, i_k)$ is reached, each interval (x_j, x_{j+1}) is allocated to agent i_j, with the cut points themselves given arbitrarily.*

For convenience, we have defined strictly mediated protocols as a separate model rather than as a special case of Robertson-Webb protocols. However, given a strictly mediated protocol, it is easy to define a Robertson-Webb protocol that simulates it, so we shall also consider strictly mediated protocols as a special case of Robertson-Webb protocols.

To get some intuition, consider the following example.

Example 1. *Let $\mathcal{M}$ be some strictly mediated protocol that on an execution path reaches a leaf where the cut points discovered are $\{0.1, 0.7\}$, and the values of the agents for each subinterval are:*

- *Agent 1 has: $V_1([0, 0.1]) = v_1$, $V_1([0.1, 0.7]) = v_2$.*
- *Agent 2 has: $V_2([0, 0.1]) = w_1$, $V_2([0.1, 0.7]) = w_2$.*

Say that $\mathcal{M}$ stopped after discovering these values and allocated the subintervals in the order $[1, 2, 1]$; that is, agent 1 received $[0, 0.1] \cup [0.7, 1]$, while agent 2 received $[0.1, 0.7]$. Then $\mathcal{M}$ has the property that if the answers of the agents resulted instead in a different set of cut points, $\{x_1, x_2\}$, but the evaluate queries were answered in the same way (i.e. $V_1([0, x_1]) = v_1$, $V_1([x_1, x_2]) = v_2$, $V_2([0, x_1]) = w_1$, $V_2([x_1, x_2]) = w_2$), then $\mathcal{M}$ outputs the same allocation order (i.e. agent 1 gets $[0, x_1] \cup [x_2, 1]$ and 2 gets $[x_1, x_2]$).

The relevance of the strictly mediated model is apparent from the following lemma.

Lemma 2. *Assume there exists a strategy-proof protocol $\mathcal{M}$ for $n \geq 2$ agents with the property that there exists an outcome that is associated with a hungry valuation profile and where every agent receives a non-empty piece. Then there exists a strategy-proof strictly mediated protocol $\mathcal{R}$ with the same property.*

Proof. Rather than formally describe the protocol $\mathcal{R}$ as a decision tree, we give an informal description, from which a formal (but probably less readable) description as a decision tree could easily be derived. First, we describe the idea of the construction.

The key constraint that a strictly mediated protocol has to satisfy is to not let the sequence of queries it makes nor its final allocation depend on the exact physical location of the cut points. It can only let these actions depend on the reports of the agents. With this in mind, the idea of the protocol $\mathcal{R}$ is to directly simulate the protocol $\mathcal{M}$ step by step. But since the protocol $\mathcal{M}$ might have behavior that depends on the physical location of the cut points, we let $\mathcal{R}$ maintain a list of fictitious or pretend locations $y_t^*, t = 1, \dots, k$ in $(0, 1)$ that it feeds to $\mathcal{M}$ instead of the actual cut points $y^t, t = 1, \dots, k$ made, preserving order, i.e, with the invariant maintained that $y_t^* < y_{t'}^*$ if and only if $y_t < y_{t'}$ for all t, t'. An alternative point of view is that $\mathcal{R}$, being strictly mediated, has no precise measuring device that can determine exactly where the agents make the cut points, but that it makes its own primitive yardstick as it goes along, using the cut points actually made by the agents as marks on its yard stick. However, we also have to make sure that we preserve the outcome of $\mathcal{M}$ where all agents get a piece. Therefore, $\mathcal{R}$ has to be somewhat careful when defining the fictitious cut points.

Let X be some outcome (leaf) of $\mathcal{M}$ that is associated with a hungry value density function profile and in which all agents receives a non-empty piece. Concretely, $\mathcal{R}$ simulates $\mathcal{M}$ as described in the next cases.

Case 1: Whenever the protocol $\mathcal{M}$ wants to ask agent i a cut query Cut$(i; \alpha)$, the protocol $\mathcal{R}$ computes numbers t, t', α' and by a Cut query asks agent i to specify a point y_t in the subinterval $[y_{t'}; y_{t''}]$ between existing cut points $y_{t'}$ and $y_{t''}$ for which $V_i([y_{t'}, y_t]) = \alpha'$. The numbers t', t'', α' are computed so that a truthful agent i will execute exactly the Cut$(i; \alpha)$ query. This computation can be performed by $\mathcal{R}$ for the following reason. As we explained when we defined the Roberson-Webb model, we maintain the invariant that all new subintervals are evaluated by all agents after each Cut query in the original protocol $\mathcal{M}$. As $\mathcal{R}$ simulates $\mathcal{M}$ step by step, $\mathcal{R}$ also maintains this knowledge. When agent i returns the new cut point y_t from the Cut query, the protocol $\mathcal{R}$ needs to find a suitable fictitious cut point y_t^*. There are two sub-cases:

- In the execution of $\mathcal{M}$, it is still possible to reach X (i.e., X is a descendant of the Cut$(i; \alpha)$ node that $\mathcal{R}$ is simulating at the moment). In this case, there is a unique value for the cut point that will keep this possibility open by keeping the exeuction of $\mathcal{M}$ on the path to X. We let y_t^* be this unique value.

- In the execution of $\mathcal{M}$, it is no longer possible to reach X. In this case, we let $y_t^* = (y_{t'}^* + y_{t''}^*)/2$.

In both sub-cases, we feed y_t^* back to $\mathcal{M}$ as the fictitious answer to the Cut query Cut$(i; \alpha)$.

Case 2: Whenever $\mathcal{M}$ asks agent i an Eval query Eval$(i; y_{t'}^*)$ where $y_{t'}(y_{t'}^*)$ is a new real (fictitious) cut point, $\mathcal{R}$ asks agent i to evaluate $[y_{t''}, y_{t'}]$ where y'' is the largest cut point smaller than $y_{t'}$ (or 0, if no such cut point exists). As $y_{t''}$ is an older cut point, $\mathcal{R}$ already knows a report for $V_i([0, y_{t''}])$ and can return a report for $V_i([0, y_{t''}])$ as the sum of these reports to $\mathcal{M}$.

Case 3: Finally, when $\mathcal{M}$ makes an allocation in the end, $\mathcal{R}$ allocates each subinterval $[y_{t'}, y_{t''}]$ to the agent to which $\mathcal{M}$ allocates $(y_{t'}, y_{t''})$.

Now we check that $\mathcal{R}$ has the desired properties:

◦ By construction, $\mathcal{R}$ is strictly mediated.
◦ By construction, there is an outcome of $\mathcal{R}$ associated with a hungry valuation profile where all agents get a piece of the cake, namely the hungy value density function profile where agents answer Eval queries in the way that keeps the execution of $\mathcal{M}$ on the track to X.
◦ Finally, suppose that $\mathcal{R}$ is not truthful. That is, there is a scenario where an agent, say agent 1, has value density function v, and there is a strategies σ_i for agents $i = 2, .., n$ in $\mathcal{R}$, so that truthful reporting is not an optimal strategy for agent 1. Then some other strategy π is strictly better, yielding an increase in payoff $\delta > 0$. We claim that then there is a value density function v' for agent 1 and strategies σ_i' for agents $i = 2, .., n$ in $\mathcal{M}$ so that truthful reporting is not an optimal strategy for agent 1. Hence $\mathcal{M}$ is also not truthful, contradicting the assumption on $\mathcal{M}$. We define:

- v' simply to be any value density function that is consistent with the reports that $\mathcal{M}$ receives by $\mathcal{R}$ when $\mathcal{R}$ is given input v and
- σ_i' to be the strategy of reporting to $\mathcal{M}$ the way $\mathcal{R}$ reports to $\mathcal{M}$ for agent i, when agent i plays according to σ_i in $\mathcal{R}$.

Then truthful reporting of v' is not optimal for agent 1 in $\mathcal{M}$, if all agents $i = 2, .., n$ play according to σ_i, since agent 1 would get an increase in payoff of δ by playing the strategy π' of reporting to $\mathcal{M}$ the way $\mathcal{R}$ reports to $\mathcal{M}$ for agent 1, when agent 1 plays using π in $\mathcal{R}$. □

Our next lemma shows that strategy-proof strictly mediated protocols are very restricted in their behavior.

Lemma 3. *Let $\mathcal{M}$ be a strictly mediated protocol for $n \geq 2$ agents that has some outcome, with an associated hungry valuation profile, in which each agent receives a non-empty piece. Then $\mathcal{M}$ is not strategy-proof.*

Proof. Let X be a leaf of the protocol, associated with a hungry value density function profile $\mathbf{v}$, in which an allocation is made where all the agents receive a non-empty piece. Denote by $x_1 < x_2 < \cdots < x_M$ the labels of the cut points defined on the path to X. (Note that they were not necessarily

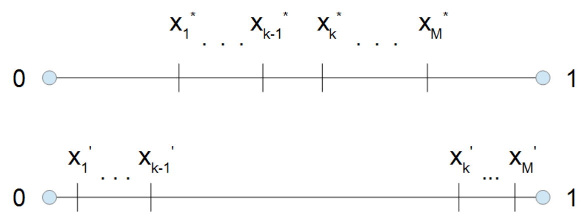

Figure 2: Valuation profile $\mathbf{v}^*$ at the top and $\mathbf{v}'$ at the bottom. The cut points $x_1^*, \ldots, x_M^*$ are completely contained in the interval (x_{k-1}^*, x_k').

defined in that order on the path - the indices here indicate the order of the cut points according to usual ordering of real numbers. Also note that since the protocol is strictly mediated, $x_1, \ldots, x_M$ are symbolic label names rather than actual real numbers.) Without loss of generality, assume that $\mathcal{M}$ asks the first Cut query to agent 1. Since the allocation at X is non-dictatorial, then we have the following:

- agent 1 does not receive the entire cake
- agent 1 receives at least one subinterval, say (x_{k-1}, x_k).

Suppose any concrete sequence $x_1^* < .. \cdots .. < x_M^*$ of real numbers strictly between 0 and 1 is given. By continously deforming $\mathbf{v}$, we can construct a hungry value density function profile $\mathbf{v}^*$ associated with the leaf X so that when the protocol is executed on $\mathbf{v}^*$, the actual cut point with label x_i becomes x_i^*. That is, the allocation order (from left to right) computed by $\mathcal{M}$ is the same for both valuations, $\mathbf{v}^*$ and $\mathbf{v}'$ – for example, if agent 2 gets the piece $[0, x_1^*]$ on input $\mathbf{v}^*$, then agent 2 also gets the piece $[0, x_1']$ on input $\mathbf{v}'$, and viceversa.

We shall define two such valuation profiles (see Figure 2), namely $\mathbf{v}^*$ (with actual cut points $x_1^* < .. \cdots .. < x_M^*$) and $\mathbf{v}'$ (with actual cut points $x_1' < \cdots < x_M'$). We choose $\mathbf{v}^*$ to be an arbitrary hungry profile associated with X. Let w^* be the valuation of agent 1 for his piece in the outcome associated with $\mathbf{v}^*$; since agent 1 does not receive the entire cake at X, we have that $w < 1$. The profile $\mathbf{v}'$ is constructed and its cut points are chosen so that $x_{k-1}' < x_1^* < \cdots < x_M^* < x_k'$. Moreover, x_{k-1}' is chosen sufficiently close to 0 and x_k' sufficiently close to 1, to ensure that the valuation of agent 1 for (x_{k-1}, x_k) according to $\mathbf{v}^*$ is strictly larger than w. Since $\mathbf{v}'$ is associated with X, agent 1 gets the subinterval (x_{k-1}', x_k') when both agents report according to $\mathbf{v}'$.

Consider now the following strategy σ_i for each other agent $i \in \{2, \ldots, n\}$:

1. If agent 1 answers the first cut query according to a valuation consistent with $\mathbf{v}'$, then agent i answers for the remainder of the protocol as if his valuation is also consistent with $\mathbf{v}'$.
2. Otherwise, agent i answers truthfully throughout the protocol.

Observe that since the first Cut query is addressed to agent 1, the answer of the agent will be different under valuations $\mathbf{v}^*$ and $\mathbf{v}'$ by choice of the two profiles.

Suppose the true type profile of the agents is $\mathbf{v}^*$ and agents $2, \ldots, n$ adopt strategies $\sigma_2, \ldots, \sigma_n$, respectively. Then, if agent 1 answers truthfully, it gets in the end a piece worth w to the agent. However, if agent 1 lies by answering according to a valuation consistent with $\mathbf{v}'$, then in the end it gets a piece for which (x_{k-1}, x_k) is a subset. Since the value of agent 1 for this interval alone is strictly larger than w^*, $\mathcal{M}$ is not strategy-proof. □

We are now ready to prove our two main theorems.

Proof. (of Theorem 1) Suppose we have a strategy-proof protocol $\mathcal{M}$ which is not a dictatorship when restricted to two hungry agents. We have by Lemma 1 that it non-trivially shares the cake between the agents in all outcomes corresponding to truthful reports of hungry value density functions. By Lemma 2, there is a strategy-proof strictly mediated protocol with an outcome in which the cake is shared. But this contradicts Lemma 3. □

Proof. (of Theorem 2) Suppose we have a strategy-proof protocol $\mathcal{M}$ with some outcome where $n \geq 3$ agents get a non-empty piece. By Lemma 2, there exists a strategy-proof strictly mediated protocol $\mathcal{R}$ with the same property. This contradicts Lemma 3. □

4 Randomized Protocols

In this section we turn to randomized protocols in the Robertson-Webb model. A randomized protocol can formally be defined similar to the definition of deterministic protocols in Section 2, except that the decision tree now contains three types of internal nodes: cut nodes, evaluate nodes, and chance nodes. The cut and evaluate nodes are the same as for deterministic protocols, while each chance node X has some number of directed outgoing edges, each of which is labeled with the probability of being taken when the execution reaches the node X.

A protocol $\mathcal{M}$ is *truthful in expectation* if on every instance, the expected utility of an agent i (taken over all random coin tosses of $\mathcal{M}$) is the best possible when behaving truthfully, regardless of the strategies of the other agents. Mossel and Tamuz [2010] showed a randomized *direct revelation* protocol that is truthful in expectation and computes a *perfect* allocation, that is, an allocation $A = (A_1, \ldots, A_n)$ where $V_i(A_j) = 1/n, \forall i, j \in N$:

> *Given as input valuations $V_1, \ldots, V_n$, find a perfect partition $A = (A_1, \ldots, A_n)$ and allocate it using a random permutation π over $\{1, \ldots, n\}$ (i.e. agent i receives the piece A_{π_i}).*

Perfect partitions are guaranteed to exist and require at most $n(n-1)^2$ cuts [Alon, 1987], so the protocol is well-defined, but not constructive. Here, we observe that we can "discretize" the Mossel-Tamuz protocol to get an *explicit* Robertson-Webb protocol that is truthful in expectation and computes an "almost perfect" allocation.

Theorem 3. *Given $\varepsilon > 0$, there is a randomized Robertson-Webb protocol $\mathcal{M}_\epsilon$ that asks at most $O(n^2/\varepsilon)$ queries, is truthful in expectation and allocates to each agent a piece of value between $1/n - \epsilon$ and $1/n + \epsilon$, according to the valuation functions of all agents.*

```
K ← ⌈2n(n−1)/ε⌉
for each agent i ∈ {1, ..., n} do
  x_{i,0} ← 0
  x_{i,K+1} ← 1
  for each j ∈ {1, ..., K} do
    x_{i,j} ← Cut(i; j/K)
  end for
end for
X ← ⋃_{i=1}^{n} {x_{i,1}, ..., x_{i,K}}
for each subset Y ⊆ X, with |Y| ≤ n(n − 1) do
  for each allocation (A_1, ..., A_n) definable by cuts in Y
  do
    for each i, j ∈ {1, ..., n} do
      n_{i,j} ← #{k ∈ {0, ..., K} | (x_{i,k}, x_{i,k+1}) ⊆ A_j}
      w_{i,j} ← (1/K) · n_{i,j}
    end for
    if (1/n − 2/K ≤ w_{i,j}) and (w_{i,j} ≤ 1/n + 2/K), for all i, j
    then
      π ← RandomPermutation ({1, ..., n})
      for each agent i ∈ {1, ..., n} do
        W_{π_i} ← A_i // Agent π(i) gets piece A_i
      end for
      return W
    end if
  end for
end for
```

Algorithm 2: Randomized Robertson-Webb protocol that is truthful in expectation and almost perfect

Proof. Given $\varepsilon > 0$, let $\mathcal{M}_\varepsilon$ be the protocol in Algorithm 4. At a high level, protocol $\mathcal{M}_\varepsilon$ asks each agent to divide the cake in many small cells (K of them) of equal value $1/K$; then $\mathcal{M}_\varepsilon$ exhaustively enumerates all subsets Y of size bounded by $n(n-1)$ from the cut points supplied by the agents. Given that a perfect partition is guaranteed to exist on the continuous cake within at most $n(n-1)$ cuts [Alon, 1987], one of the sets Y is guaranteed to work. That is, $\mathcal{M}_\varepsilon$ finds a set of points Y and an allocation A that uses exclusively cut points in Y such that:

- every point in Y is close to a cut point of a perfect partition $\bar{A}$ on the continuous cake (within distance at most $1/K$ from the point of view of each agent)
- the allocation order (from left to right) in A is the same as the one in $\bar{A}$.

Then for each contiguous piece $X \in A$, the value of an agent i for X is the same as agent i's value for the corresponding piece $\bar{X}$ in the perfect partition $\bar{A}$, except possibly for a gain or loss of $2/K$ due to estimation errors (at most $1/K$ at each endpoint of X) It follows that A approximates $\bar{A}$ within an error of at most ε. Finally, once $\mathcal{M}_\varepsilon$ finds an appropriate partition, it allocates it using a random permutation π, and so the expected value of each agent is *exactly* $1/n$, regardless of the strategies of the other agents, as in the Mossel-Tamuz protocol. Thus $\mathcal{M}_\varepsilon$ is truthful in expectation and ε-perfect. □

5 Discussion

As stated in the introduction, an important open question is whether the impossibility theorems hold with respect to the stronger notion of Nash equilibrium. Leaving the Robertson-Webb model, the question of whether non-dictatorial and *fair* direct revelation protocols exist for hungry agents (e.g. with piecewise constant value density functions) remains open as well. A partial answer was given for the class of piecewise uniform valuations [Chen *et al.*, 2013], but these valuations do not capture hungry agents and, moreover, the protocol of Chen *et al.* may discard cake.

References

[Alon, 1987] N. Alon. Splitting necklaces. *Advances in Mathematics*, 63:241–253, 1987.

[Aumann *et al.*, 2013] Y. Aumann, Y. Dombb, and A. Hassidim. Computing socially-efficient cake divisions. In *Proc. 12th AAMAS*, pages 343–350, 2013.

[Bei *et al.*, 2012] X. Bei, N. Chen, X. Hua, B. Tao, and E. Yang. Optimal proportional cake cutting with connected pieces. In *Proc. 26th AAAI*, pages 1263–1269, 2012.

[Brams and Taylor, 1996] S. J. Brams and A. D. Taylor. *Fair Division: From Cake-Cutting to Dispute Resolution*. Cambridge University Press, 1996.

[Brams *et al.*, 2012] S. J. Brams, M. Feldman, J. Morgenstern, J. K. Lai, and A. D. Procaccia. On maxsum fair cake divisions. In *Proc. 26th AAAI*, pages 1285–1291, 2012.

[Brânzei and Miltersen, 2013] S. Brânzei and P. B. Miltersen. Equilibrium analysis in cake cutting. In *Proc. 12th AAMAS*, pages 327–334, 2013.

[Brânzei *et al.*, 2013] S. Brânzei, A. D. Procaccia, and J. Zhang. Externalities in cake cutting. In *Proc. 23rd IJCAI*, pages 55–61, 2013.

[Caragiannis *et al.*, 2011] I. Caragiannis, J. K. Lai, and A. D. Procaccia. Towards more expressive cake cutting. In *Proc. 22nd IJCAI*, pages 127–132, 2011.

[Chen *et al.*, 2013] Y. Chen, J. K. Lai, D. C. Parkes, and A. D. Procaccia. Truth, justice, and cake cutting. *GEB*, 77:284–297, 2013.

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

[Cohler *et al.*, 2011] Y. J. Cohler, J. K. Lai, D. C. Parkes, and A. D. Procaccia. Optimal envy-free cake cutting. In *Proc. 25th AAAI*, pages 626–631, 2011.

[Edmonds and Pruhs, 2006a] J. Edmonds and K. Pruhs. Balanced allocations of cake. In *Proc. 47th FOCS*, pages 623–634, 2006.

[Edmonds and Pruhs, 2006b] J. Edmonds and K. Pruhs. Cake cutting really is not a piece of cake. In *Proc. 17th SODA*, pages 271–278, 2006.

[Gibbard, 1973] A. Gibbard. Manipulation of voting schemes. *Econometrica*, 41:587–602, 1973.

[Gutman and Nisan, 2012] A. Gutman and N. Nisan. Fair allocation without trade. In *Proc. 11th AAMAS*, pages 719–728, 2012.

[Kash *et al.*, 2013] I. Kash, A. D. Procaccia, and N. Shah. No agent left behind: Dynamic fair division of multiple resources. In *Proc. 12th AAMAS*, pages 351–358, 2013.

[Kurokawa *et al.*, 2013] D. Kurokawa, J. K. Lai, and A. D. Procaccia. How to cut a cake before the party ends. In *Proc. 27th AAAI*, pages 555–561, 2013.

[Maya and Nisan, 2012] A. Maya and N. Nisan. Incentive compatible two player cake cutting. In *Proc. 8th WINE*, pages 170–183, 2012.

[Mossel and Tamuz, 2010] E. Mossel and O. Tamuz. Truthful fair division. In *Proc. 3rd SAGT*, pages 288–299, 2010.

[Nisan, 2007] Noam Nisan. *Algorithmic Game Theory*, chapter 9, pages 209–241. Cambridge University Press, New York, NY, USA, 2007.

[Procaccia, 2009] A. D. Procaccia. Thou shalt covet thy neighbor's cake. In *Proc. 21st IJCAI*, pages 239–244, 2009.

[Procaccia, 2013] A. D. Procaccia. Cake cutting: Not just child's play. *CACM*, 56(7):78–87, 2013.

[Robertson and Webb, 1998] J. M. Robertson and W. A. Webb. *Cake Cutting Algorithms: Be Fair If You Can*. A. K. Peters, 1998.

[Satterthwaite, 1975] M. Satterthwaite. Strategy-proofness and Arrow's conditions. *JET*, 10:187–217, 1975.

[Steinhaus, 1948] H. Steinhaus. The problem of fair division. *Econometrica*, 16:101–104, 1948.

[Woeginger and Sgall, 2007] G. J. Woeginger and J. Sgall. On the complexity of cake cutting. *Discrete Optimization*, 4:213–220, 2007.