Truth, Justice, and Cake Cutting*

Yiling Chen† John K. Lai‡ David C. Parkes§ Ariel D. Procaccia¶

Superman: “I’m here to fight for truth, justice, and the American way.”
Lois Lane: “You’re gonna wind up fighting every elected official in this country!”

Superman (1978)

Abstract

Cake cutting is a common metaphor for the division of a heterogeneous divisible good. There are numerous papers that study the problem of fairly dividing a cake; a small number of them also take into account self-interested agents and consequent strategic issues, but these papers focus on fairness and consider a strikingly weak notion of truthfulness. In this paper we investigate the problem of cutting a cake in a way that is truthful and fair, where for the first time our notion of dominant strategy truthfulness is the ubiquitous one in social choice and computer science. We design both deterministic and randomized cake cutting mechanisms that are truthful and fair under different assumptions with respect to the valuation functions of the agents.

1 Introduction

Cutting a cake is often used as a metaphor for allocating a divisible good. The difficulty is not cutting the cake into pieces of equal size, but rather that the cake is not uniformly tasty: different agents prefer different parts of the cake, depending, e.g., on whether the toppings are strawberries or cookies. The goal is to divide the cake in a way that is “fair”; the definition of fairness is a nontrivial issue in itself, which we discuss in the sequel. The cake cutting problem dates back to the work of Steinhaus in the 1940s [17], and for over sixty years has attracted the attention of mathematicians, economists, political scientists, and more recently, computer scientists [9, 10, 15].

Slightly more formally, the cake is represented by the interval $[0, 1]$. Each of n agents has a valuation function over the cake, which assigns a value to every given piece of cake and is additive.

*A significantly shorter version of this paper appeared in the proceedings of AAAI 2010. Here we discuss the group strategyproofness of our mechanism, fill in missing proofs, and provide an extended discussion of related work and future directions. The paper was also presented in the Third International Workshop on Computational Social Choice in Düsseldorf, Germany, September 2010, at a workshop on prior-free mechanism design in Guanajuato, Mexico, May 2010, and in the Harvard EconCS seminar, February 2010.

†Harvard SEAS, 33 Oxford Street, Cambridge, MA 02138, email: yiling@eecs.harvard.edu.
‡Harvard SEAS, 33 Oxford Street, Cambridge, MA 02138, email: jklai@post.harvard.edu.
§Harvard SEAS, 33 Oxford Street, Cambridge, MA 02138, email: parkes@eecs.harvard.edu.
¶Harvard SEAS, 33 Oxford Street, Cambridge, MA 02138, email: arielpro@seas.harvard.edu.

The goal is to find a partition of the cake among the agents (while possibly throwing a piece away) that satisfies one or several fairness criteria. In this paper we consider the two most prominent criteria. A *proportional* allocation is one where the value each agent has for its own piece of cake is at least $1/n$ of the value it assigns to the entire cake. An *envy-free (EF)* allocation is one where the value each agent assigns to its own piece of cake is at least as high as the value it assigns to any other agent's piece of cake. There is a rather large body of literature on fairly cutting a cake according to these two criteria (see, e.g., the books by Robertson and Webb [16] and Brams and Taylor [6]).

So far we have briefly discussed "justice", but have not yet mentioned "truth." Taking the game-theoretic point of view, an agent's valuation function is its private information, which is reported to a cake cutting mechanism. We would like a mechanism to be *truthful*, in the sense that agents are motivated to report their true valuation functions. Like fairness, this idea of truthfulness also lends itself to many interpretations. One variation, referred to as *strategy-proofness* in previous papers by Brams et al. [4, 5], assumes that an agent will report its true valuation rather than lie if there *exist* valuations of the other agents such that reporting truthfully yields at least as much value as lying. In the words of Brams et al., "...the players are risk-averse and never strategically announce false measures if it does not guarantee them more-valued pieces. ... Hence, a procedure is strategy-proof if no player has a strategy that dominates his true value function." [5, page 362].

The foregoing notion is strikingly weak compared to the notion of truthfulness that is common in the social choice literature. Indeed, strategy-proofness is usually taken to mean that an agent can *never* benefit by lying, that is, *for all* valuations of the other agents reporting truthfully yields at least as much value as lying. Put another way, truth-telling is a dominant strategy. This notion is worst-case, in the sense that an agent cannot benefit by lying even if it is fully knowledgeable of the valuations of the other agents. In order to prevent confusion we will avoid using the term "strategy-proof," and instead refer to the former notion of Brams et al. as "weak truthfulness" and to the latter standard notion as "truthfulness."

To illustrate the difference between the two notions, consider the most basic cake cutting mechanism for the case of two agents, the *Cut and Choose* mechanism.[1] Agent 1 cuts the cake into two pieces that are of equal value according to its valuation; agent 2 then chooses the piece that it prefers, giving the other piece to agent 1. This mechanism is trivially proportional and EF.[2] It is also weakly truthful, as if agent 1 divides the cake into two pieces that are unequal according to its valuation then agent 2 may prefer the piece that is worth more to agent 1. Agent 2 clearly cannot benefit by lying. The mechanism is even truthful (in the strong sense) if the agents have the same valuation function. However, the mechanism is not truthful in general. Indeed, consider the case where agent 1 would simply like to receive as much cake as possible, whereas the single-minded agent 2 is only interested in the interval $[0, \epsilon]$ where ϵ is small (for example, it may only be interested in the cherry). If agent 1 follows the protocol it would only receive half of the cake. Agent 1 can do better by reporting that it values the intervals $[0, \epsilon]$ and $[\epsilon, 1]$ equally, since then it would end up with almost the entire cake by choosing to cut pieces $[0, \epsilon], [\epsilon, 1]$.

In this paper we consider the design of truthful and fair cake cutting mechanisms. We will design mechanisms where the agents report valuations and the mechanism, taking the role of a direct revelation mechanism, determines the cuts of the cake. However, there is a major obstacle

[1]This mechanism is described here with the agents taking actions; equivalently, the mechanism acts on behalf of agents using the reported valuations.

[2]Proportionality and envy-freeness coincide if there are two agents and the entire cake is allocated.

that must be circumvented: regardless of strategic issues, and when there are more than five agents, even finding a proportional and EF allocation in a bounded number of steps with a deterministic mechanism is a long-standing open problem! See Procaccia [15] for an up-to-date discussion.[3] We shall therefore restrict ourselves to specific classes of valuation functions where efficiently finding fair allocations is a non-issue; the richness of our problem stems from our desire to additionally achieve truthfulness.

In particular, we will focus on the case where agents hold *piecewise uniform* and *piecewise linear* valuation functions. Piecewise uniform valuation functions capture the case where each agent is interested in a collection of subintervals of $[0, 1]$ and has the same marginal value for each fractional piece in each subinterval. While these valuations are restrictive, they are expressive enough to capture some realistic settings. Piecewise uniform valuations capture the case when some parts of the cake satisfy a certain property that an agent cares about and an agent desires as much of these parts as possible. An example of such a setting is a shared computing server. Each researcher has certain times during which the server is not useful because of scheduling constraints or waiting times for experiments, but outside of these times, the researcher would like as much time on the computing server as possible. Piecewise linear valuation functions capture the case where an agent's marginal value is linear, and are significantly more expressive. In particular, these valuation functions can approximate any valuation function.

Our results. In Section 3 we consider deterministic mechanisms when agent valuations are piecewise uniform. Our main result is a deterministic mechanism for any number of agents that is truthful, proportional, EF, and polynomial-time when agents have piecewise uniform valuations.

To gain intuition for our mechanism, it is insightful to examine the special case of two agents. In the two agent mechanism, the crux of the allocation is how we should allocate intervals that both agents desire. We can throw away intervals that are undesired by any agent, and allocate the intervals that are only desired by a single agent to that agent. Our mechanism then allocates the intervals desired by both agents to try to equalize the total lengths given to each agent. If the length of intervals only desired by agent i exceeds the length of intervals only desired by agent j, then agent i will be given a smaller share of the mutually desired intervals. To see how this encourages truthfulness, consider the deviation where an agent misstates and increases the lengths of intervals only desired by that agent. The agent receives all of these intervals (which include intervals it does not actually desire), but obtains a smaller share of the mutually desired intervals. It turns out that the equalization of lengths removes the incentive for agents to misstate their valuations.

For the special case of two agents, our mechanism also corresponds to a natural process where agents perform a series of Pareto-improving swaps. The initial allocation grants each agent exactly one half of the desirable parts of the cake. In such an allocation, an agent will be given pieces of cake for which it has no value. The agents then swap equal lengths of these pieces with each other, increasing overall welfare since the agents receive pieces which provide positive value in exchange for pieces which provide no value. Once these mutually beneficial swaps are exhausted, we find swaps that improve an agent's welfare while maintaining the other agent's welfare. When all beneficial swaps have been exhausted, the mechanism terminates. For this special case, our mechanism starts with a fair allocation, and sequentially improves agent welfare through these swaps, and in the process, eliminates the incentive to misstate valuations. In the two agent case, there is a clear sequence of swaps to make; however, when we extend to more than two agents, it is unclear what

[3] To be precise, previous work assumed that the entire cake has to be allocated, but this does not seem to be a significant restriction in the context of fairness.

the proper sequence of swaps is and the intuition of Pareto-improving trades does not naturally extend.

In the general mechanism for n agents, our mechanism examines what the fairness requirement means for the allocation given to different agents. Agents who receive a small amount of the cake due to fairness requirements are handled first and given the best possible fair allocation. As an example of agents who might be handled first, there could be a set of agents who all desire the same small interval of the cake. For fairness to be maintained, each of these agents must receive the same fraction of this small interval. These agents would be selectively handled earlier by the mechanism. By protecting the disadvantaged agents in this way, the mechanism removes the incentive to report larger intervals than those that are truly desired.

In Section 4 we consider randomized mechanisms. We slightly relax truthfulness by asking that the mechanism be *truthful in expectation*, that is, an agent cannot hope to increase its expected value by lying for any reports of other agents. For general valuations, we present a simple randomized mechanism that is truthful in expectation, and is *ex-post* proportional and EF. Our result relies on the existence of *perfect partitions*, which are partitions of the cake where every agent has value exactly $1/n$ for every element of the partition. Given a perfect partition, the randomized mechanism constructs an allocation by indexing the elements of the partition, and allocating the elements to agents based on a randomly drawn permutation. The perfect partition guarantees fairness, while truthfulness in expectation is preserved because the agent's expected value is a function of the agent's value for the entire cake and not of the specific partition.

While perfect partitions are known to exist, there is no known algorithm for constructing perfect partitions for general valuations. By focusing on the relatively weak assumption that agents hold *piecewise linear* valuation functions, we are able to give an explicit construction of perfect partitions and show that such a construction can be implemented in polynomial time.

Related work. We have recently learned of independent parallel work by Mossel and Tamuz, who ask similar questions about truthful and fair cake cutting [14]. However, their work only considers randomized mechanisms, and they provide existence results rather than concrete mechanisms for specific classes of valuations. They also focus solely on proportionality rather than proportionality and envy-freeness together. Under general assumptions on valuation functions (which are essentially the same as our definition of valid valuations), they show that there exists a mechanism that is truthful in expectation and always guarantees each agent a value of more than $1/n$. The results are then extended to the case of indivisible goods. The technical overlap between the two papers is very small; we refer the reader's attention to this overlap in a footnote in Section 4.

Thomson [18] showed that a truthful and Pareto-optimal mechanism must be dictatorial in the slightly different setting of pie-cutting. In pie cutting, the pie is modeled as a circular object and the feasible cuts are wedges, while in cake cutting the cake is modeled as the interval [0,1] and feasible cuts are subintervals. Though the two settings appear similar and even possibly equivalent, they are actually distinct and results for one setting do not readily carry over to the other. Brams et. al. [5] provide a more in depth discussion. Note that Pareto-optimality is not a fairness property and neither implies, nor is implied by, envy-freeness or proportionality.

Our deterministic mechanism is closely related to a method proposed by Bogomolnaia and Moulin [3] in the context of the *random assignment problem*, and the network flow techniques we employ in our analysis generalize the reinterpretation of this method in terms of network flow due to Katta and Sethuraman [11].

To understand the Bogomolnaia and Moulin [3] setting, consider a set of indivisible items that must be assigned to agents. In the random assignment problem, items can be assigned to agents randomly, i.e., a random assignment is a probability distribution over deterministic assignments. Each agent may only receive one item in a deterministic assignment, though the agent may have preferences over which items it prefers.

Bogomolnaia and Moulin [3] study the random assignment problem when the agents have dichotomous preferences over the items. For each agent, the set of items can be partitioned into acceptable and unacceptable items (where all acceptable items have value 1 and all unacceptable items have value 0). They provide a random assignment method, called the *egalitarian assignment solution*, and show that it is truthful, EF, and satisfies other highly desirable properties such as efficiency. In this setting, it turns out that combining envy-freeness with efficiency implies proportionality, although this is not explicitly discussed by the authors.[4]

The cake cutting problem under piecewise uniform valuation functions is similar to a random assignment problem, as one can mark the beginning and end of each agent's desired intervals and treat the subintervals between consecutive marks as items. A random assignment that gives an item to an agent with probability p can be interpreted as assigning a p-fraction of the item to the agent. However, there are two fundamental differences with our setting. First, in our setting agents are interested in receiving as much of their desired "items" as possible (rather than just one item). Second, in our setting dichotomous preferences would mean that agents value all desired subintervals equally, which is clearly not the case since these subintervals have different lengths.[5] Nevertheless, it turns out that the egalitarian assignment solution is very similar to a special case of our truthful, deterministic mechanism under this strong assumption. Katta and Sethuraman [11] observe that the egalitarian assignment solution can be computed in polynomial time using network flow techniques, so the connections we establish to these techniques are an independent generalization of this observation. Interestingly, it is noted in Katta and Sethuraman [11] that the egalitarian assignment solution is identical to another independent mechanism for finding a lexicographically optimal flow in a network due to Megiddo [13].

In earlier work, Bogomolnaia and Moulin [2] study random assignments under *strict* ordinal preferences, and propose a solution that satisfies a weaker notion of truthfulness (which does not imply truthfulness in our setting) as well as envy-freeness and other properties. In terms of the agents' preferences this setting is incomparable to ours since our agents may be indifferent between subintervals, and cannot hold arbitrary ordinal preferences over subintervals between consecutive marks, since if two agents desire two subintervals, both agents would value the longer subinterval more than the shorter.

The results of Bogomolnaia and Moulin [2] were extended by Katta and Sethuraman [11] to the full preference domain where agents need not have dichotomous preferences and can be indifferent

[4] As discussed in Bogomolnaia and Moulin [3], with dichotomous preferences, one can view the random assignment problem as a matching problem where one side of the market finds all matches acceptable. Efficiency demands that a random assignment be a lottery over efficient and individually rational matchings. Efficient and individually rational matchings can be characterized using the Gallai-Edmonds decomposition. If an agent is not matched with probability 1, then the agent must be under-demanded in the Gallai-Edmonds decomposition. This means the agent desires only over-demanded items, which will be allocated with probability 1. If there are n agents and agent i receives a desirable item with probability less than $1/n$, then there has to be some other agent who receives items in agent i's desirable set with probability greater than $1/n$, contradicting envy-freeness. This line of argument is similar to how we prove proportionality for our deterministic mechanism and why envy-freeness implies proportionality without free disposal.

[5] In general no discretization of the cake would necessarily yield subintervals of equal length that correspond to dichotomous preferences. If we assume that desired intervals have rational endpoints then such a discretization can be found, but the number of subintervals would be exponentially large, leading to computational intractability.

between objects. Katta and Sethuraman [11] establish that in this more general setting even Bogomolnaia and Moulin's [2] weaker notion of truthfulness is in fact incompatible with envy-freeness and the additional efficiency requirement of *ordinal efficiency*.[6]

The mechanism that they propose satisfies the last two properties and hence does not even satisfy their weaker notion of truthfulness. The authors do not discuss proportionality, but the concept is difficult to define for the full preference domain. Agents are no longer indifferent over all desirable items, and receiving a desirable item with probability $1/n$ yields different utilities depending on the specific random assignment.

In summary, there has been related work on the random assignment problem which uses techniques similar to ours, but the cake cutting setting is fundamentally different and cannot be viewed as a special case of the random assignment problem. Even if the cake were *ex-ante* discretized into indivisible items, in cake cutting agents can be allocated multiple items and desire as many items as possible. In prior work on the random assignment problem, for the most closely related case of dichotomous preferences, there are truthful, fair, and efficient mechanisms. When non-dichotomous strict preferences are allowed, fair and efficient mechanisms exist which satisfy a weaker notion of truthfulness. When non-dichotomous and non-strict preferences are allowed, even this weaker notion of truthfulness cannot be satisfied if the mechanism is EF and ordinally efficient. Our assumption of piecewise uniform valuations for our deterministic mechanism is similar in spirit to dichotomous preferences, and it may be that allowing more general valuations leads to impossibility results, paralleling the work on the random assignment problem. Even in the case of piecewise uniform valuations, our deterministic mechanism is non-trivial, and it does not seem possible to apply similar ideas to more general valuations. It may well be that piecewise uniform valuations are at the frontier of what is possible for truthful and fair cake cutting mechanisms.

2 Preliminaries

We consider a heterogeneous cake, represented by the interval $[0, 1]$. A *piece of cake* is a *finite* union of subintervals of $[0, 1]$. We sometimes abuse this terminology by treating a piece of cake as the set of the (inclusion-maximal) intervals that it contains. The length of the interval $I = [x, y]$, denoted $\text{len}(I)$, is $y - x$. For a piece of cake X we denote $\text{len}(X) = \sum_{I \in X} \text{len}(I)$.

The set of agents is denoted $N = \{1, \ldots, n\}$. Each agent $i \in N$ holds a private valuation function V_i, which maps given pieces of cake to the value agent i assigns them. Formally, each agent i has a *value density function*, $v_i : [0, 1] \to [0, \infty)$, that is piecewise continuous. The function v_i characterizes how agent i assigns value to different parts of the cake. The value of a piece of cake X to agent i is then defined as

$$V_i(X) = \int_X v_i(x)dx = \sum_{I \in X} \int_I v_i(x)dx.$$

We note that the valuation functions are *additive*, i.e. for any two disjoint pieces X and Y, $V_i(X \cup Y) = V_i(X) + V_i(Y)$, and *non-atomic*, that is $V_i([x, x]) = 0$ for every $x \in [0, 1]$. The last

[6] Because Katta and Sethuraman [11] examine the full preference domain, the notion of ordinal efficiency is used to compare two random assignments. From a single agent's perspective, an assignment P dominates an assignment Q if the probability of the agent receiving one of its top k items (including ties) is greater in P than in Q for every k. An assignment Q is ordinally efficient if there is no other assignment P that weakly dominates Q for all agents and strictly dominates Q for at least one agent.

property implies that we do not have to worry about the boundaries of intervals, i.e., open and closed intervals are identical for our purposes. We further assume that the valuation functions are *normalized*, i.e. $V_i([0,1]) = \int_0^1 v_i(x)dx = 1$.

A cake cutting mechanism is a function f from the valuation function of each agent to an allocation $(A_1, \ldots, A_n)$ of the cake such that the pieces A_i are pairwise disjoint. For each $i \in N$ the piece A_i is allocated to agent i, and the rest of the cake, i.e., $[0,1] \setminus \bigcup_{i \in N} A_i$, is thrown away. Here we are assuming *free disposal*, that is, the mechanism can throw away resources without incurring a cost.

We say that an allocation $A_1, \ldots, A_n$ is *proportional* if for every $i \in N$, $V_i(A_i) \geq 1/n$, that is, each agent receives at least a $(1/n)$-fraction of the cake according to its own valuation. We say that an allocation is *envy-free (EF)* if for every $i, j \in N$, $V_i(A_i) \geq V_i(A_j)$, i.e., each agent prefers its own piece of cake to the piece of cake allocated to any other agent. A proportional (resp., EF) cake cutting mechanism always returns a proportional (resp., EF) allocation.

Note that when $n = 2$ proportionality implies envy-freeness. Indeed, $V_i(A_i) + V_i(A_{3-i}) \leq 1$, and hence if $V_i(A_i) \geq 1/2$ then $V_i(A_{3-i}) \leq 1/2$. Under the free disposal assumption the converse is not true. For example, an allocation that throws away the entire cake is EF but not proportional. In general, when $n > 2$ proportionality neither implies nor is implied by envy-freeness.[7]

A cake cutting mechanism f is *truthful* if when an agent lies it is allocated a piece of cake that is worth, according to its real valuation, no more than the piece of cake it was allocated when reporting truthfully. Formally, denote $A_i = f_i(V_1, \ldots, V_n)$, and let $\mathcal{V}$ be a class of valuation functions. The mechanism f is truthful if for every agent i, every collection of valuations functions $V_1, \ldots, V_n \in \mathcal{V}$, and every $V_i' \in \mathcal{V}$, it holds that $V_i(f_i(V_1, \ldots, V_n)) \geq V_i(f_i(V_1, \ldots, V_{i-1}, V_i', V_{i+1}, \ldots, V_n))$.

3 Deterministic Mechanisms and Piecewise Uniform Valuations

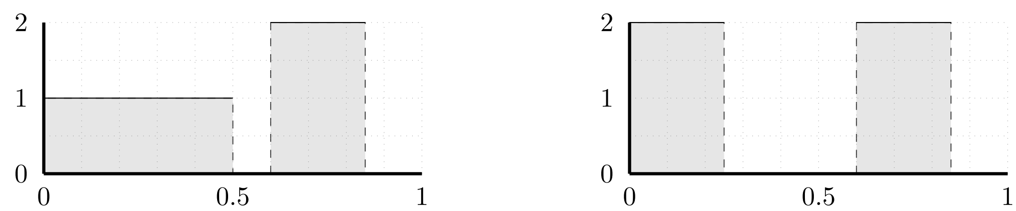
In order to attain a truthful, deterministic mechanism, we focus on a restricted family of valuation functions. The restricted family is still rich enough to make the problem challenging and capture real world situations, and we believe that our results provide a foundation for future work on truthful cake cutting. Indeed, it remains an open question whether truthful, deterministic mechanisms exist for richer families of valuations.

We say that a valuation function V_i is *piecewise constant* if its corresponding value density function v_i is piecewise constant, that is $[0,1]$ can be partitioned into a finite number of intervals such that v_i is constant on each interval (see Figure 1(a)). We say that V_i is *piecewise uniform* if moreover v_i is either some constant $c \in \mathbb{R}_+$ (the same one across intervals) or zero. See Figure 1(b) for an illustration.

Piecewise uniform valuation functions imply that agent $i \in N$ is uniformly interested in a finite union of intervals, which we call its *reference piece of cake* and denote by U_i. For example, in Figure 1(b), $U_i = [0, 0.25] \cup [0.6, 0.85]$. Given a piece of cake X, it holds that $V_i(X) = \text{len}(X \cap U_i)/\text{len}(U_i)$. From the computational perspective, the size of the input to the cake cutting mechanism is the number of bits that define the boundaries of the intervals in the agents' reference pieces of cake.

In the rest of this section we assume that the valuation functions are piecewise uniform. We believe that piecewise uniform valuations are very natural. An agent would have such a valuation

[7] If free disposal is not assumed, that is, the entire cake is allocated, then envy-freeness implies proportionality for any n.



(a) Value density function for a piecewise constant valuation that is not piecewise uniform.

(b) Value density function for a piecewise uniform valuation.

Figure 1: An illustration of special value density functions.

function if it is simply interested in pieces of the good that have a certain property, e.g., a child only likes portions of the cake that have chocolate toppings, and wants as much cake with chocolate toppings as possible. As a more practical example, assume that the cake represents access time to a backup server. Each agent is equally interested in time intervals when its computer is idle or when it is not modifying its data. We investigate more general valuations in Section 4.

3.1 A deterministic mechanism

Before introducing our mechanism we present some required notation. Let $S \subseteq N$ be a subset of agents and let X be a piece of cake. Let $D(S, X)$ denote the portions of X that are valued by at least one agent in S. Formally, $D(S, X) = \left(\bigcup_{i \in S} U_i\right) \cap X$, and is itself a union of intervals.

Let $\text{avg}(S, X) = \text{len}(D(S, X))/|S|$ denote the average length of intervals in X desired by at least one agent in S. We say that an allocation is *exact* with respect to S and X if it allocates to each agent in S a piece of cake of length $\text{avg}(S, X)$ comprised *only* of desired intervals. Clearly this requires allocating all of $D(S, X)$ since the total length of allocated intervals is $\text{avg}(S, X) \cdot |S| = \text{len}(D(S, X))$. Suppose $S = \{1, 2\}$ and $X = [0, 1]$: if $U_1 = U_2 = [0, 0.2]$ then agents 1 and 2 receiving $[0, 0.1]$ and $[0.1, 0.2]$ respectively is an exact allocation; but if $U_1 = [0, 0.2], U_2 = [0.3, 0.7]$ then there is no exact allocation.

The deterministic mechanism for n agents with piecewise uniform valuations is a recursive mechanism that finds a subset of agents with a certain property, makes the allocation decision for that subset, and then makes a recursive call on the remaining agents and the remaining intervals. Specifically, for a given set of agents $S \subseteq N$ and a remaining piece of cake to be allocated X, we find the subset $S' \subseteq S$ of agents with the smallest $\text{avg}(S', X)$. We then give an exact allocation of $D(S', X)$ to S'. We recurse on $S \setminus S'$ and the intervals not desired by any agent in S', i.e. $X \setminus D(S', X)$. The pseudocode of the mechanism is given as Mechanism 1.

In particular, Steps 2 and 3 of SubRoutine imply that if $S = \{i\}$ then $A_i = D(S, X)$. For example, suppose $X = [0, 1]$, $U_1 = [0, 0.1]$, $U_2 = [0, 0.39]$, and $U_3 = [0, 0.6]$. In this case, the subset with the smallest average is $\{1\}$, so agent 1 receives all of $[0, 0.1]$ and we recurse on $\{2, 3\}, [0.1, 1]$. In the recursive call, set $\{2\}$ has average $0.39 - 0.1 = 0.29$, set $\{3\}$ has average $0.6 - 0.1 = 0.5$, and set $\{2, 3\}$ has average $(0.6 - 0.1)/2 = 0.25$. As a result, the entire set $\{2, 3\}$ is chosen as the set with smallest average, and an exact allocation of $[0.1, 1.0]$ is given to agents 2 and 3. One possible allocation is to give agent 2 $[0.1, 0.35]$ and agent 3 $[0.35, 0.6]$. Note that if agent 1 uniformly values $[0, 0.2]$ instead, the first call would choose $\{1, 2\}$ as the subset with the smallest average, equally allocating $[0, 0.39]$ between agents 1 and 2 and giving the rest, $[0.39, 0.6]$, to agent 3.

Mechanism 1 $(V_1, \ldots, V_n)$

1. SubRoutine$(\{1, \ldots, n\}, [0, 1], (V_1, \ldots, V_n))$

SubRoutine$(S, X, V_1, \ldots, V_n)$:

1. If $S = \emptyset$, return.
2. Let $S_{\min} \in \underset{S' \subseteq S}{\text{argmin}} \text{ avg}(S', X)$ (breaking ties arbitrarily).
3. Let $E_1, \ldots, E_n$ be an exact allocation with respect to $S_{\min}, X$ (breaking ties arbitrarily). For each $i \in S_{\min}$, set $A_i = E_i$.
4. Subroutine$(S \setminus S_{\min}, X \setminus D(S_{\min}, X), (V_1, \ldots, V_n))$.

Our goal in the rest of this section is to prove the following theorem.

Theorem 3.1. *Assume that the agents have piecewise uniform valuation functions. Then Mechanism 1 is truthful, proportional, EF, and polynomial-time.*

To prove the theorem we exploit a connection to network flow in Section 3.1.2. As explained in Section 1, this technique is a simple generalization of related results in the economics and operations research literature, but we include some of the details for completeness. Our main technical contributions in this section are the truthfulness and fairness of Mechanism 1, which are established in Section 3.1.4.

3.1.1 An analysis of the two agent mechanism

To gain intuition for the general case of Theorem 3.1 we first give a thorough and somewhat informal exposition of the case of two agents. Note that designing truthful, proportional and EF mechanisms even for this case is nontrivial. To see the difficulty, consider an intuitive first attempt at a proportional and EF mechanism. We have already seen that Cut and Choose is not truthful for two agents. Another straightforward approach would be to mark the end points of all submitted intervals and divide every resulting subinterval equally between the two agents. One possibility is to always give the left half of each subinterval to agent 1 and the right half to the agent 2. This mechanism is clearly proportional and envy-free since every agent receives value exactly 1/2. However, it is not truthful due to a simple example. Under this mechanism, if both agents value the entire cake, agent 1 receives $[0, 0.5]$ and the agent 2 receives $[0.5, 1]$. Suppose that agent 1's true valuation consists of only $[0, 0.5]$. If it reports truthfully, it receives $[0, 0.25], [0.5, 0.75]$ which gives value 0.5. The agent can gain by instead reporting that it values all of $[0, 1]$ and receive $[0, 0.5]$ which gives it value 1. In particular, suppose that when both agents report $[0, 1]$, agent 1 receives $[0, 0.5]$. In order for the mechanism to be truthful, whenever agent 1 reports some set of subintervals of $[0, 0.5]$ and agent 2 reports $[0, 1]$, the mechanism must allocate to agent 1 all of this agent's desired intervals.

We now discuss our truthful Mechanism 1 for the case of two agents. Recall that U_i denotes the reference piece of cake of agent $i \in N$, and for $i \in \{1, 2\}$ let $W_i = U_i \setminus U_{3-i}$, and $W_{12} = U_1 \cap U_2$. Note that W_1, W_2, W_{12} are disjoint, and this will allow us to interchangeably write $\text{len}(W_1) + \text{len}(W_2) + \text{len}(W_{12})$ and $\text{len}(W_1 \cup W_2 \cup W_{12})$.

For two agents, the decision of the mechanism hinges on the choice of $S_{\min}$ in the first call to SubRoutine. There are three possibilities for $S_{\min}$: $\{1\}, \{2\}, \{1, 2\}$. If $\{i\}$ is the set with the smallest average then agent i receives $U_i = W_i \cup W_{12}$ and agent $3 - i$ receives $U_{3-i} \setminus U_i = W_{3-i}$. On the other hand, if $\{1, 2\}$ is the subset with the smallest average, then the mechanism gives an exact allocation with each agent receiving equal lengths. Agent i receives all of W_i and W_{12} is split so that the total lengths for each agent are equal.

To prove that the mechanism for two agents is well-defined, fair, and truthful, it is useful to introduce variables $\delta_i = (\text{len}(W_{12}) - \text{len}(W_i) + \text{len}(W_{3-i}))/2$ for each agent. Note that $\delta_1 + \delta_2 = \text{len}(W_{12})$ and $\delta_1 + \text{len}(W_1) = \text{len}(W_{12} \cup W_1 \cup W_2)/2 = \delta_2 + \text{len}(W_2)$. Qualitatively, δ_i measures the share of W_{12} that agent i is entitled to due to the size of W_i. Smaller values of $\text{len}(W_i)$ give the agent a larger claim to W_{12} and increase δ_i. We can relate values of these variables to the choices of $S_{\min}$ made by the mechanism.

By definition of W_i and W_{12}, we have that $\text{avg}(\{1\}) = \text{len}(W_1 \cup W_{12})$, $\text{avg}(\{2\}) = \text{len}(W_2 \cup W_{12})$, $\text{avg}(\{1, 2\}) = \text{len}(W_1 \cup W_2 \cup W_{12})/2$. There are three cases:

1. $\delta_1 < 0, \delta_2 > \text{len}(W_{12})$. Since $\text{len}(W_1) + \delta_1 = \text{len}(W_2) + \delta_2$, it must be that $\text{len}(W_1) > \text{len}(W_2)$ and therefore $\text{avg}(\{1\}) > \text{avg}(\{2\})$. Moreover, using $\delta_2 > \text{len}(W_{12})$, we have that

$$\frac{\text{len}(W_{12}) - \text{len}(W_2) + \text{len}(W_1)}{2} > \text{len}(W_{12}),$$

   and it follows that

$$\frac{\text{len}(W_{12}) + \text{len}(W_2) + \text{len}(W_1)}{2} > \text{len}(W_{12}) + \text{len}(W_2),$$

   that is, $\text{avg}(\{1, 2\}) > \text{avg}(\{2\})$.

   Thus, this case corresponds to $\{2\}$ being the subset with the smallest average, and the mechanism gives all of $W_2 \cup W_{12}$ to agent 2 and all of W_1 to agent 1.

2. $0 \leq \delta_1 \leq \text{len}(W_{12}), 0 \leq \delta_2 \leq \text{len}(W_{12})$. We can use the same arguments as the previous case with the inequalities flipped to show that $\text{avg}(\{1, 2\}) < \text{avg}(\{1\})$ and $\text{avg}(\{1, 2\}) < \text{avg}(\{2\})$. This case corresponds to $\{1, 2\}$ being the subset with the smallest average, and the mechanism gives an exact allocation to agents 1 and 2. Recall that an exact allocation of piece X to agents S requires each agent to be given desired intervals of length exactly $1/|S|$ of $\text{len}(D(S, X))$. In this case, $D(S, X) = W_1 \cup W_2 \cup W_{12}$, so each agent needs to be given exactly $\text{len}(W_1 \cup W_2 \cup W_{12})/2$ in desired intervals. Since the total length of desired intervals is $\text{len}(W_1 \cup W_2 \cup W_{12})$, we cannot waste any intervals by giving them to the wrong agent. Thus, W_1 goes to agent 1 and W_2 goes to agent 2. All that remains is to allocate W_{12}. Recalling that $\delta_1 + \text{len}(W_1) = \delta_2 + \text{len}(W_2)$, an exact allocation requires that agent 1 be given δ_1 of W_{12} and agent 2 be given δ_2 of W_{12}.

3. $\delta_2 < 0, \delta_1 > \text{len}(W_{12})$. The analysis is the same as case 1, with agents 1 and 2 changing roles. This corresponds to the case where $\{1\}$ is the subset with the smallest average. Agent 1 receives all of $W_1 \cup W_{12}$ and agent 2 receives all of W_2.

An interpretation of the two agent mechanism. When $|S| = 2$, the mechanism is equivalent to a swapping procedure. As before, let W_1, W_2, W_{12} describe the intervals that only agent 1 desires, only agent 2 desires, and both agents desire. Discard the intervals that neither agent desires, and give an initial grant of half of W_1, W_2, W_{12} to each agent. Assume without loss of generality that $\text{len}(W_1) \leq \text{len}(W_2)$. The swapping procedure can be described as follows.

1. Swap pieces X, Y of equal length where agent 1 owns X, agent 2 owns Y, $X \subseteq W_2$, $Y \subseteq W_1$.
2. Swap pieces X, Y of equal length where agent 1 owns X, agent 2 owns Y, $X \subseteq W_2$, $Y \subseteq W_{12}$.
3. If there are still pieces of W_2 owned by agent 1, give these intervals to agent 2.

This procedure first gives each agent an equal piece of each kind of interval. The first swaps performed are mutually beneficial: each agent gives pieces of the other agent's desired intervasl in exchange for pieces of its own desired intervals. However, eventually, agent 1 obtains all of W_1 and these swaps no longer exist. The second swaps involve trades where agent 2 receives remaining pieces of W_2 in exchange for giving up pieces of W_{12}. This trade does not improve the utility of agent 2, but does improve the utility of agent 1. If agent 2 gives away all of its share of W_{12}, then agent 1 receives all of W_1 and W_{12}, and agent 2 is given all of W_2 due to step 3 which gives the remaining shares of W_2 to agent 2 for free. If agent 2 obtains all of W_2 without relinquishing all of its share of W_{12}, then agents 1 and 2 split W_{12} (potentially unequally), and each agent receives W_i.

In fact, we can give conditions for when each of the two cases occurs. After the first set of swaps, agent 1 has all of W_1, and agent 2 has $\text{len}(W_2 \cup W_1)/2$ of W_2 and still desires $(\text{len}(W_2) - \text{len}(W_1))/2$ of W_2. If $\text{len}(W_{12})/2 \geq (\text{len}(W_2) - \text{len}(W_1))/2$ then agent 2 can get all of W_2 without exhausting all of its share of W_{12}, and the agents equally split $W_1 \cup W_2 \cup W_{12}$. Otherwise, agent 1 receives all of W_1, W_{12} and agent 2 receives only W_2.

The crux of the swapping procedure is the allocation of W_{12}. Each agent always receives W_i, and W_{12} is allocated to try and equalize the total lengths of intervals obtained by each agent. However, this may not always be possible because agent 1 may desire less than $\text{len}(W_1 \cup W_2 \cup W_{12})/2$, and in this case, agent 1 receives all of W_{12}.

To summarize, there are two cases in the swapping procedure:

1. $\text{len}(W_1 \cup W_{12}) \geq \text{len}(W_1 \cup W_2 \cup W_{12})/2$. Agent 1 receives W_1, agent 2 receives W_2 and the agents split W_{12} so that their allocated lengths are equal.
2. $\text{len}(W_1 \cup W_{12}) < \text{len}(W_1 \cup W_2 \cup W_{12})/2$. Agent 1 receives $W_1 \cup W_{12}$ and agent 2 receives W_2.

To see that this swapping procedure is exactly equivalent to Mechanism 1 for two agents, assume again without loss of generality that $\text{len}(W_1) \leq \text{len}(W_2)$. In Mechanism 1, either $\{1\}$ or $\{1, 2\}$ is the subset with the smallest average. $\{1, 2\}$ is the chosen subset if $\text{len}(W_1) + \text{len}(W_{12}) \geq (\text{len}(W_1 \cup W_2 \cup W_{12})/2$. If $\{1, 2\}$ is chosen, then agents 1 and 2 split $W_1 \cup W_2 \cup W_{12}$ in an exact allocation. If $\{1\}$ is chosen, then agent 1 receives all of W_1, W_{12}, and agent 2 receives all of W_2.

Properties of the two agent mechanism. The interpretation of the mechanism as a swapping procedure immediately implies that Mechanism 1 is EF and proportional. Indeed, the two agents are granted initial allocations that are worth exactly 1/2 to each agent, and subsequent swaps cannot decrease an agent's utility. This interpretation also implies that the mechanism is well-defined, i.e., when $S_{min} = \{1, 2\}$ then it is possible to give an exact allocation with respect to S_{min}, X.

In order to establish Theorem 3.1 for the two agent case it remains to show truthfulness. We will take the point of view of agent $i \in \{1, 2\}$. If it holds that $\delta_i > \text{len}(W_{12})$ then the agent receives all desired intervals and has no incentive to deviate, hence we can assume that $\delta_i \leq \text{len}(W_{12})$.

Note that an agent always receives all of W_i, so profitable manipulations will try to obtain more of W_{12} by increasing δ_i while not losing too much of W_i. Also note that by definition of δ_i, an increase or decrease of $\text{len}(W_i)$ by k will respectively decrease or increase δ_i by $k/2$.

Suppose that agent i reports $U'_i \neq U_i$, inducing new pieces W'_{12}, W'_1, W'_2. Before manipulating the agent receives $\text{len}(W_i) + \max(\delta_i, 0)$. Note that

$$\text{len}(W'_{12}) + \text{len}(W'_{3-i}) = \text{len}(U_{3-i}) = \text{len}(W_{12}) + \text{len}(W_{3-i}),$$

that is, this sum is not affected by the report of agent i.

If $\text{len}(W'_1) = \text{len}(W_1)$ then δ_i is unchanged, and the agent receives the same length of intervals (though the actual intervals received may not be desired). If $\text{len}(W'_i) = \text{len}(W_i) - k$ then the agent loses at least k of W_i and gains at most $k/2$ from an increase in δ_i; this is not profitable. Finally, if $\text{len}(W'_i) = \text{len}(W_i) + k$, then the agent gains undesired intervals of length k and δ_i is weakly smaller, so the agent does not gain any more of W_{12} by deviating.

3.1.2 Exact allocations and maximum flows

Having defined Mechanism 1 and shown that it has the desired properties in the case of two agents, we now generalize the proofs to n agents. Before turning to properties of truthfulness and fairness, we point out that so far it is unclear whether Mechanism 1 is well-defined. In particular, the mechanism requires an exact allocation E with respect to the subset $S_{\min}$ and X, but it remains to show that such an allocation exists, and to provide a way to compute it. To this end we exploit a close relationship between exact allocations and maximum flows in networks.

For a given set of agents $S \subseteq N$ and a piece of cake to be allocated X, define a graph $G(S, X)$ as follows. We keep track of a set of marks, which will be used to generate nodes in $G(S, X)$. First mark the left and right boundaries of all intervals that are contained in X. For each agent $i \in N$ and subinterval in U_i, mark the left and right boundaries of subintervals that are contained in $U_i \cap X$. When we have finished this process, each pair of consecutive markings will form an interval such that each agent either uniformly values the entire interval or values none of the interval. In $G(S, X)$, create a node for each interval I formed by consecutive markings, and add a node for each agent $i \in N$, a source node s, and a sink node t. For each interval I, add a directed edge from source s to I with capacity equal to the length of the interval. Each agent node is connected to t by an edge with capacity $\text{avg}(S, X)$. For each interval-agent pair (I, i), add a directed edge with infinite capacity from node I to the agent i if agent i desires interval I.

For example, suppose $U_1 = [0, 0.25] \cup [0.5, 1]$ and $U_2 = [0.1, 0.4]$. If $X = [0, 1]$ then the interval markings will be $\{0, 0.1, 0.25, 0.4, 0.5, 1\}$. Agent 1 values $[0, 0.1]$, both agents value $[0.1, 0.25]$, agent 2 values $[0.25, 0.4]$, neither agent values $[0.4, 0.5]$ and agent 1 values $[0.5, 1]$. It holds that $\text{len}(D(\{1, 2\}, [0, 1])) = 0.9$. Average values are 0.75, 0.3 and 0.45 for sets $\{1\}, \{2\}$ and $\{1, 2\}$ respectively. See Figure 2 for an illustration of the induced flow network.

Lemma 3.2. *Let $S \subseteq N$, and let X be a piece of cake. There is a flow of size len$(D(S, X))$ in $G(S, X)$ if and only if for all $S' \subseteq S$, avg$(S', X) \geq$ avg(S, X).*

We prove the lemma using an application of the classic Max-Flow Min-Cut Theorem (see, e.g., [8]).

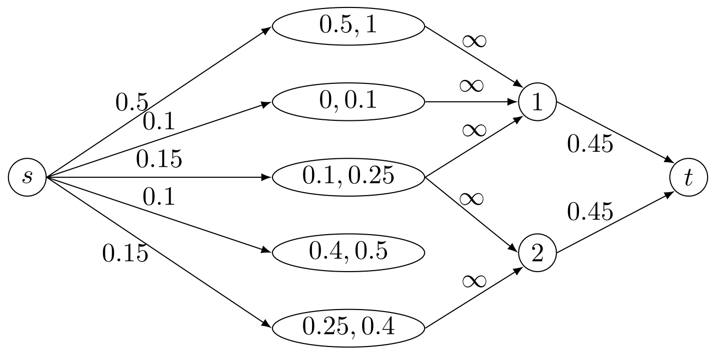


Figure 2: The flow network induced by the example.

Proof. Assume that for all $S' \subseteq S$, $\text{avg}(S', X) \geq \text{avg}(S, X)$. By the Max-Flow Min-Cut Theorem, the minimum capacity removed from a graph in order to disconnect the source and sink is equal to the size of the maximum flow. The only edges with finite capacity in $G(S, X)$ are the ones that connect agent nodes to the sink, and the ones that connect the source to the interval nodes.

Construct a candidate minimum cut by disconnecting some set of agent nodes $T \subseteq S$ from the sink at cost $|T| \cdot \text{avg}(S, X)$ and then disconnecting all the (s, I) connections to interval nodes I desired by an agent $i \in S \setminus T$. This means that the total additional capacity we need to remove is $\text{len}(D(S \setminus T, X))$, the total length of intervals desired by at least one agent in $S \setminus T$. By assumption, this is at least $|S \setminus T| \cdot \text{avg}(S, X)$. As a result, this cut has capacity of at least

$$|T| \cdot \text{avg}(S, X) + |S \setminus T| \cdot \text{avg}(S, X) = |S| \cdot \text{avg}(S, X) = \text{len}(D(S, X)).$$

In the other direction, assume that there is a flow of size $\text{len}(D(S, X))$ in $G(S, X)$, and assume for contradiction that there exists $S' \subseteq S$ such that $\text{avg}(S', X) < \text{avg}(S, X)$. Construct a cut by disconnecting the (s, I) connections to interval nodes desired by an agent $i \in S'$, and disconnecting the agent nodes $S \setminus S'$ from the sink. The total capacity of the cut is

$$|S'| \cdot \text{avg}(S', X) + |S \setminus S'| \cdot \text{avg}(S, X) < \text{len}(D(S, X)),$$

and by the Max-Flow Min-Cut Theorem the maximum flow must be of size less than $\text{len}(D(S, X))$, in contradiction to our assumption. □

The following lemma establishes that a flow of size $\text{len}(D(S, X))$ in $G(S, X)$ induces an exact allocation.

Lemma 3.3. *Let $S \subseteq N$, and let X be a piece of cake. There exists an exact allocation with respect to S, X if and only if there exists a maximum flow of size len$(D(S, X))$ in $G(S, X)$.*

Proof. Suppose that we have a maximum flow of size $D(S, X)$; we show how to use this flow to generate an exact allocation with respect to S, X. For each edge between interval node I and agent $i \in N$ that receives positive flow of c in the max flow, allocate c of I to agent i. This allocation is feasible because the interval nodes represent disjoint subintervals of X and the flow to the interval's node is limited by the capacity of the edge between s and the interval node, which is the length of the interval. In addition, this allocation must give each agent exactly $\text{avg}(S, X)$ in desired intervals. To see this, note that all paths to the sink must pass through agent nodes, and the sum of capacities of the edges between between the agents and the sink is $D(S, X)$. For a maximum flow to have size $D(S, X)$, these edges must saturated.

In the other direction, suppose that we have an exact allocation with respect to S, X. We can generate a feasible flow of size $\text{len}(D(S, X))$ by setting a flow of c on an edge (I, i) if agent i receives c of interval I in the exact allocation, and saturating all the edges (s, I) and (i, t) for intervals I and agents $i \in N$. □

By combining the "if" directions of Lemma 3.2 and Lemma 3.3 we see that the mechanism is indeed well-defined: if S has the smallest average then there exists an exact allocation with respect to S, X.[8] Moreover, we obtain a tractable mechanism for computing an exact allocation, by computing the maximum flow and deriving an exact allocation. A maximum flow can be computed in time that is polynomial in the number of nodes, that is, polynomial in our input size (see, e.g., [8]).

3.1.3 Polynomial time

In order to show that Mechanism 1 can be implemented in polynomial time it remains to show that it is also possible to implement Step 2 of SubRoutine in polynomial time. Indeed, an efficient implementation of Step 2 would mean that SubRoutine can be implemented in polynomial time, and there are at most $n+1$ calls to SubRoutine. So, the task is to find $S_{min} \in \text{argmin}_{S' \subseteq S} \text{avg}(S, X)$ in polynomial time, given $S \subseteq N$ and a piece of cake X. This can be done using network flow arguments, which are an easy variation on those employed by Katta and Sethuraman [11] (see Section 1 above). For completeness we quickly describe an implementation that is less efficient than [11] but nevertheless polynomial time.

Given $S \subseteq N$, a piece of cake X, and $c > 0$, construct a graph $G'(S, X, c)$; this graph is identical to $G(S, X)$ as defined above, except that the capacity on the edges between the agents and the sink is c (instead of $\text{avg}(S, X)$). The proof of the following statement is identical to the proof of Lemma 3.2 (by replacing $\text{avg}(S, X)$ with c everywhere): there is a flow of size $c|S|$ in the network $G'(S, X, c)$ if and only if for all $S' \subseteq S$, $\text{avg}(S', X) \geq c$.

Assume that the boundaries of the agents' reference pieces of cake are represented by at most k bits. There is a maximum c^* such that $G'(S, X, c^*)$ has a flow of size $c^*|S|$. This c^* is a member of a set K of numbers in $[0, 1]$ that includes all the possible values of $\text{avg}(S', X)$ for $S' \subseteq S$. It holds that $|K|$ is exponential in k and n,[9] and by doing a binary search on K we can find c^* in polynomial time.

Now, let $c^*+\alpha$ be the minimum element of K that is larger than c^*, and consider $G'(S, X, c^*+\alpha)$. This network does not have a network flow of size $(c^* + \alpha)|S|$. We can find a minimum cut in this network in polynomial time (see, e.g., [8]). In this cut there is a subset $T \subseteq S$ such that the cut separates $S \setminus T$ from the sink, and the intervals desired by T from the source. This subset T must have $\text{avg}(T, X) < c^* + \alpha$, and by the construction of K this means that $\text{avg}(T, X) \leq c^*$.

We wish to claim that T is the S_{min} we are looking for. Indeed, note that since there is a flow of size $c^*|S|$ in $G'(S, X, c^*)$, it must hold that for all $S' \subseteq S$, $\text{avg}(S', X) \geq c^*$, which directly implies the claim.

[8] Note that the network in Figure 2 does not satisfy the minimum average requirement and does not provide a corresponding exact allocation.

[9] $\text{len}(D(S', X))$ is upperbounded by 1, and therefore clearly has a number of possible values that is exponential in k. Since $\text{avg}(S', X) = \text{len}(D(S', X))/|S'|$, in order to enumerate all the possible values of $\text{avg}(S', X)$ it is sufficient to enumerate the values of $\text{len}(D(S', X))$ divided by $|S'|$, for each possible value of S'.

3.1.4 Truthfulness and fairness

Our main tool in proving that Mechanism 1 is truthful, proportional and EF is the following lemma.

Lemma 3.4. *Let $S_1, \ldots, S_m$ be the ordered sequence of agent sets with the smallest average as chosen by Mechanism 1 and $X_1, \ldots, X_m$ be the ordered sequence of pieces to be allocated in calls to* SubRoutine. *That is, $X_1 = [0,1], X_2 = X_1 \setminus D(S_1, X_1), \ldots, X_m = X_{m-1} \setminus D(S_{m-1}, X_{m-1})$. Then for all $i > j$, avg$(S_i, X_i) \geq$ avg(S_j, X_j), and agents that are members of later sets receive weakly more in desired lengths.*

Proof. Suppose not. Then at some point, $\text{avg}(S_i, X_i) > \text{avg}(S_{i+1}, X_{i+1})$. Now consider $S_i \cup S_{i+1}$. We will show that $\text{avg}(S_i \cup S_{i+1}, X_i) < \text{avg}(S_i, X_i)$, contradicting the choice of S_i as the subset of agents with the smallest avg at step i. Note that S_i and S_{i+1} are disjoint since agents are removed once they have been part of the subset with the smallest average. Thus,

$$
\begin{aligned}
\text{avg}(S_i \cup S_{i+1}) &= \frac{\text{len}(D(S_i \cup S_{i+1}, X_i))}{|S_i| + |S_{i+1}|} \\
&= \frac{\text{len}(D(S_i, X_i)) + \text{len}(D(S_{i+1}, X_i \setminus D(S_i, X_i)))}{|S_i| + |S_{i+1}|} \\
&= \frac{|S_i| \cdot \text{avg}(S_i, X_i) + |S_{i+1}| \cdot \text{avg}(S_{i+1}, X_{i+1})}{|S_i| + |S_{i+1}|} \\
&< \frac{|S_i| \cdot \text{avg}(S_i, X_i) + |S_{i+1}| \cdot \text{avg}(S_i, X_i)}{|S_i| + |S_{i+1}|} \\
&= \text{avg}(S_i, X_i),
\end{aligned}
$$

where the second transition is true since S_i and S_{i+1} are disjoint, and the inequality follows from our assumption. □

Envy-freeness. Envy-freeness now follows immediately from Lemma 3.4. Indeed, consider an agent $i \in N$, and as before let S_j be the subset of agents with the smallest average in the j'th call to SubRoutine. Suppose $i \in S_j$. The agent does not envy other agents in S_j since these agents are given an exact allocation and all receive the same length in desired intervals. By Lemma 3.4, the agent does not envy agents in S_k for $k < j$ because the amount agents receive weakly increases with each call. The agent does not envy agents in S_k for $k > j$ because all intervals desired by the agent are allocated and removed from consideration when the agent receives its allocation.

Proportionality. We establish proportionality by proving a simple lemma that states that all desired intervals are allocated.

Lemma 3.5. *Mechanism 1 assigns all subintervals desired by at least one agent in S to some agent in S.*

Proof. We prove this by showing that SubRoutine always allocates desired intervals to some agent. We proceed by strong induction on the number of agents in the call to SubRoutine. In the base case where $|S| = 1$, we give the agent all desired intervals, so this trivially satisfies the required property. For the inductive step, if S' is the subset of agents with the smallest average, an exact allocation of $D(S', X)$ is made. Recall that an exact allocation gives $\text{len}(D(S', X))/|S'|$ in desired intervals to each agent. For this to be possible, all of $D(S', X)$ must be given to some agent. Therefore, the required property is satisfied on the allocation made to the subset of agents with the smallest average. Strong induction shows that the required property is satisfied on $X \setminus D(S', X)$, completing the proof. □

The lemma implies that we only dispose of intervals that are not desired by any agent. Combining Lemma 3.5 with envy-freeness gives us proportionality. Indeed, suppose that some agent i receives less than $\text{len}(D(\{i\}, [0, 1])/|S|$ of its desired intervals. Because no desired intervals are thrown away, some other agent must receive at a length of at least $\text{len}(D(\{i\}, [0, 1])/|S|$ of the desired intervals of agent i, in contradiction to envy-freeness.

Truthfulness. We establish truthfulness by strong induction on the number of agents in the call to SubRoutine. In the base case, we just have a single agent. SubRoutine in this case gives the agent all desired intervals in the piece of cake to be allocated, so the agent has no incentive to deviate. In the inductive step, there are two possible deviations. An agent can report a valuation which changes the choice of set of agents with the smallest average, or the agent can report a valuation which does not change the smallest average set but changes the exact allocation selected for the smallest average set.

We first consider deviations which try to modify the exact allocation for S, X. In this case, under truthful reports, each agent is allocated exactly $\text{len}(D(S, X))/|S|$. Suppose the agent deviates, causing the set of desired intervals to change to $D'(S, X)$. If $\text{len}(D'(S, X)) \leq \text{len}(D(S, X))$, then this is not profitable for the agent since it receives intervals totaling a weakly smaller length. Suppose then that $\text{len}(D'(S, X)) > \text{len}(D(S, X))$. Now each agent receives $\text{len}(D'(S, X))/|S|$ in desired intervals. Since the other agents are assumed to be truthful, the allocation assigns these agents intervals they truly desire (this may not be true for the deviating agent). The sum total of all intervals truthfully desired by agents in S is $D(S, X)$. Subtracting off the intervals allocated to other agents leaves the maximum length of remaining desired intervals for agent i. Since $\text{len}(D(S, X)) - (|S| - 1)D'(S, X)/|S| \leq \text{len}(D(S, X))/|S|$, this deviation is not profitable.

We next consider deviations which attempt to change the choice of subset with the smallest average. There are two cases to consider for a deviating agent $i \in N$. Let S_{min} denote the subset with the smallest average chosen under truthful reports, and let avg' denote the new averages induced by a deviation by the agent.

1. $i \notin S_{min}$. Note that for any set S', if $i \notin S'$, agent i cannot change $D(S', X)$ with its reports. As a result, the agent cannot make the mechanism choose some other $S', i \notin S'$ since the agent cannot change $\text{avg}(S_{min}, X)$ or $\text{avg}(S', X)$. Therefore, the agent's only deviation is to try and make the mechanism choose a set $S', i \in S'$. However, in order to do so, the agent must make $\text{avg}'(S', X) \leq \text{avg}(S_{min}, X)$ and will receive $\text{avg}'(S', X)$ in desired lengths. By Lemma 3.4, the agent was receiving at least $\text{avg}(S_{min}, X)$ under truthful reports since the agent was chosen in a later round, so this is not profitable. Truthfulness then follows by strong induction since the agent has no incentive to change S_{min} and cannot change the piece of cake $X \setminus D(S_{min}, X)$ allocated in the recursive call.

2. $i \in S_{min}$. The agent receives $\text{avg}(S_{min}, X)$ under truthful reports. Suppose the agent deviates and forces selection of another set S' as the set with the smallest average. If $i \in S'$ then to be profitable, $\text{avg}'(S', X) \geq \text{avg}(S_{min}, X)$. If $i \notin S'$, then agent i could not have affected

avg(S', X) and S' was not chosen under truthful reports, so $\text{avg}'(S', X) \geq \text{avg}(S_{min}, X)$. Either way, to be profitable, it must be that $\text{avg}'(S', X) \geq \text{avg}(S_{min}, X)$. Now consider some agent $j \in S_{min}$. Under truthful reports, agent j was receiving $\text{avg}(S_{min}, X)$ in desired intervals. Under the potentially profitable deviation, this agent receives at least $\text{avg}'(S', X) \geq \text{avg}(S_{min}, X)$ in desired intervals by Lemma 3.4. Using arguments similar to the case above, the sum total of intervals truthfully desired by agents in S_{min} is $D(S_{min}, X)$. While agent i deviates, the other agents in S_{min} do not, so the other agents receive intervals they truly desire. This leaves

$$\text{len}(D(S_{min}, X)) - (|S_{min}| - 1) \cdot \text{avg}'(S', X) \leq \text{len}(D(S_{min}, X)) - (|S_{min}| - 1) \cdot \text{avg}(S_{min}, X)$$

in desired intervals for agent i. The right hand side equals $\text{avg}(S_{min}, X)$, so the deviation is not profitable.

This completes the proof of truthfulness. Putting everything together gives us theorem 3.1.

4 Randomized Mechanisms and Piecewise Linear Valuations

In the previous section we saw that designing deterministic truthful and fair mechanisms is not an easy task, even if the valuation functions of the agents are rather restricted. In this section we shall demonstrate that by allowing randomness we can obtain significantly more general results.

A *randomized cake cutting mechanism* outputs a random allocation given the reported valuation functions of the agents. There are very few previous papers regarding randomized mechanisms for cake cutting. A rare example is the paper by Edmonds and Pruhs [9], where they give a randomized mechanism that achieves approximate proportionality with high probability. We are looking for a more stringent notion of fairness. We say that a randomized mechanism is *universally proportional* (resp., *universally EF*) if it always returns an allocation that is proportional (resp., EF).

One could also ask for *universal truthfulness*, that is, require that an agent may never benefit from lying, regardless of the randomness of the mechanism. A universally truthful mechanism is simply a probability distribution over deterministic truthful mechanisms. However, asking for both universal fairness and universal truthfulness would not allow us to enjoy the additional flexibility that randomization provides. Therefore, we slightly relax our truthfulness requirement. Informally, we say that a randomized mechanism is *truthful in expectation* if, for all possible valuation functions of the other agents, the expected value an agent receives for its allocation cannot be increased by lying, where the expectation is taken over the randomness of the mechanism.

We remark that while truthfulness in expectation seems natural, consistent as it is with expected utility maximization, fairness (i.e., proportionality and envy-freeness) is something that we would like to hold *ex-post*; fairness is a property of the specific allocation that is being made, and continues to be relevant after the mechanism has terminated. Interestingly enough, if we were to turn this around, then achieving universal truthfulness and envy-freeness/proportionality in expectation is trivial: simply allocate the entire cake to a uniformly random agent!

4.1 A randomized mechanism

In order to design a randomized mechanism that is truthful in expectation, universally proportional, and universally EF, we consider a very special type of allocation. In the following we will not require the free disposal assumption, that is, we will consider partitions $X_1, \ldots, X_n$ of the cake such that $\bigcup_i X_i = [0,1]$. We say that a partition $X_1, \ldots, X_n$ is *perfect* if for all $i, j \in N$, $v_i(X_j) = 1/n$. Consider the following randomized mechanism.

Mechanism 2 $(V_1, \ldots, V_n)$

1. Find a perfect partition $X_1, \ldots, X_n$.
2. Draw a random permutation π over N.
3. For each $i \in N$, set $A_i = X_{\pi(i)}$.

Lemma 4.1. *Mechanism 2 is truthful in expectation, universally proportional, and universally EF.*[10]

Proof. Let us assume for now that a perfect partition exists and can be computed. The fact that the mechanism is universally proportional and universally EF follows from the definition of perfect partitions: every agent has value $1/n$ for every piece!

We turn to truthfulness in expectation. The value an agent $i \in N$ obtains by reporting truthfully is exactly $1/n$. If agent i lies then the mechanism may choose a different partition $X'_1, \ldots, X'_n$. However, for any partition $X'_1, \ldots, X'_n$ the expected value of agent i when given a random piece is

$$\sum_{j \in N} \frac{1}{n} \cdot V_i(X'_j) = \frac{1}{n} \left(\sum_{j \in N} V_i(X'_j) \right) = \frac{1}{n},$$

where the second equality follows from the fact that the valuation functions are additive. $\square$

Finding perfect partitions. Lemma 4.1 holds much promise, in that it is valid for all valuation functions. But there still remains the obstacle of actually finding a perfect partition given the valuation functions of the agents. Does such a partition exist, and can it be computed? More than two decades ago, Noga Alon [1] proved that if the valuation functions of the agents are defined by the integral of a continuous probability measure then there *exists* a perfect partition; this is a generalization of his famous theorem on necklace splitting. Unfortunately, Alon's elegant proof is nonconstructive, and to this day there is no known constructive method under general assumptions on the valuation functions. This is not surprising since a perfect partition induces an EF allocation, and finding an EF allocation in a bounded number of steps for more than four agents is an open problem.

Lemma 4.1 nevertheless precludes a general impossibility result with respect to truthful in expectation, universally proportional, and universally EF mechanisms, unless one takes into account computational considerations. Moreover, it may be possible to find a perfect partition if we restrict our attention to specific classes of valuation functions. As an easy example, consider the class of piecewise constant valuation functions. Each agent makes a mark at the left and right boundaries of each of the intervals it is interested in, and we add two marks in 0 and 1. The value density function of each agent is constant over each subinterval between two consecutive marks, as this subinterval is either contained in one of the subintervals agent i is interested in or completely disjoint from any

[10] Mossel and Tamuz [14] make the same observation.

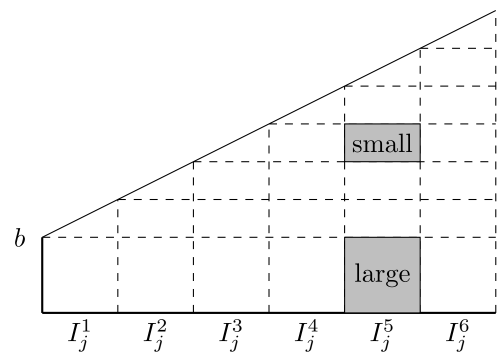


Figure 3: An illustration of the proof of Lemma 4.2. The value density function of one of the agents over an interval I_j is shown for $n = 3$. The size of each "large" rectangle is $(w/2n) \cdot b$, whereas the size of each "small" rectangle is $(w/2n) \cdot (a \cdot (w/2n)) = (w/2n)^2 \cdot a$.

of them. Hence, the allocation that assigns to each agent exactly $1/n$ of each of the subintervals between two consecutive marks is a perfect partition.

We next generalize the above observation by considering a more general class of valuation functions. A valuation function V_i is *piecewise linear* if its corresponding value density function v_i is piecewise linear on $[0, 1]$. Piecewise linear valuation functions are significantly more general than the class of piecewise constant valuation functions. A piecewise linear valuation function can be concisely represented by the intervals on which v_i is linear, and for each interval the two parameters of the linear function. The following lemma provides us with a tractable method of finding a perfect partition when the agents have piecewise linear valuation functions.

Lemma 4.2. *Assume that the agents have piecewise linear valuation functions. Consider the following procedure. We make a mark at 0 and 1, and for each agent $i \in N$ make a mark at the left and right boundaries of each interval where v_i is linear. Next, we divide each interval I_j between two consecutive marks into $2n$ consecutive and connected subintervals $I_j^1, \ldots, I_j^{2n}$ of equal length. For each such I_j and every $i \in N$ add the subintervals I_j^i and I_j^{2n-i+1} to X_i. The overall partition X_i is perfect.*

Proof. It is sufficient to show that for every interval I_j and agent $i \in N$, agent i is indifferent between the pieces $I_j^k \cup I_j^{2n-k+1}$ for every $k = 1, \ldots, n$. Let $i \in N$; note that the value density function of i on the interval I_j satisfies $v_i(x) = ax + b$ for some $a, b \in \mathbb{R}$.

Denote $w = |I_j|$. By simple geometric calculations (see Figure 3) it holds that

$$V_i(I_j^k) = \frac{w}{2n} \cdot b + \left(k - \frac{1}{2}\right) \left(\frac{w}{2n}\right)^2 \cdot a,$$

so for every $k = 1, \ldots, n$

$$V_i(I_j^k \cup I_j^{2n-k+1}) = \frac{w}{n} \cdot b + 2n \left(\frac{w}{2n}\right)^2 \cdot a,$$

which is independent of k. □

By combining Lemma 4.2 with Lemma 4.1 we obtain the following result.

Theorem 4.3. *Assume that the agents have piecewise linear valuation functions. Then there exists a randomized mechanism that is truthful in expectation, universally proportional, universally EF, and polynomial-time.*

5 Discussion

We have made progress on the new question of fair and (dominant strategy) truthful cake cutting. On the way we have made several conceptual contributions, including our consideration of restricted valuation functions in the context of cake cutting, and (independently of Mossel and Tamuz) the introduction of randomized notions of fairness and a focus on truthfulness, familiar from mechanism design but not the focus of existing cake cutting literature.

The free disposal assumption. For our deterministic result we rely heavily on the ability to throw undesired pieces of the cake away. The reason this is important is that it prevents agents from being able to gain by reporting smaller intervals, in hopes of gaining a larger share of the overlap while obtaining desired intervals for free. The following example illustrates why removing the free disposal assumption can be problematic.

Suppose there are two agents. Consider the allocation made when both agents report that they desire the interval $[0, 0.2]$. A straightforward implementation of Mechanism 1 gives $[0, 0.1]$ to agent 1 and $[0.1, 0.2]$ to agent 2. We now need to allocate $[0.2, 1]$ since we do not have free disposal. A natural way to allocate $[0.2, 1]$ is to give one contiguous half to agent 1 and the other to agent 2. Suppose that we choose to give $[0.2, 0.6]$ to agent 1 and $[0.6, 1]$ to agent 2. Though this appears benign, a new deviation has been introduced. Indeed, consider the case where agent 1 values $[0, 0.6]$ and agent 2 values $[0, 0.2]$. By reporting truthfully and applying Mechanism 1, the agent receives $[0.2, 0.6] \cup [0.8, 1]$, which it values at $2/3$. On the other hand, if agent 1 reports $[0, 0.2]$, it receives $[0, 0.1] \cup [0.2, 0.6]$, which gives value $5/6$. Notice that the problem here persists even if we allocate $[0.2, 0.6]$ to agent 2 when both agents report $[0, 0.2]$. In this case, agent 2 can deviate when it has true value $[0, 0.6]$ and agent 1 reports $[0, 0.2]$.

It is unclear whether we can dispose of the free disposal assumption. Indeed, it may be that there is an impossibility result when we force all intervals to be allocated. It would be nice to be able to apply the same insights of Mechanism 1, though as the example shows, we now have to be careful about the exact mechanics of *how* intervals are allocated rather than simply the fraction of each interval each agent receives. This adds significant complexity to the problem which we are able to avoid in Mechanism 1 when we assume free disposal.[11]

The communication model. In general, communication issues in cake cutting are rather subtle since the input (that is, the valuations functions) is continuous. To deal with this issue, Robertson and Webb [16] introduced a *concrete model of complexity* for cake cutting mechanisms; under their model a mechanism is restricted to making two types of queries, an evaluation query (whereby the mechanism learns the value of an agent with respect to a given interval) and a cut query (whereby the mechanism obtains a piece worth a given value to an agent). Through the queries the mechanism must obtain sufficient information to output a fair allocation. Existing work on

[11] Note that in Mechanism 1, any exact allocation suffices in SubRoutine, and it is not necessary to worry about the specific way an exact allocation is achieved.

computational aspects of cake cutting uses this framework (see, e.g., [7, 10, 19, 15]), but without considering truthfulness.

In this paper we have implicitly assumed that the agents report their full valuation, an assumption that facilitates our focus on strategic issues. However, the classes of valuation functions that we have considered have compact representations. Therefore, agents can in fact efficiently report their full valuation functions. Indeed, in the case of piecewise uniform valuation functions it is sufficient to report the left and right boundaries of each of the desired intervals. In the case of piecewise linear valuation functions agents also report a partition of the cake into intervals, as well as the two parameters of the linear function on each of the intervals. This allows us to design mechanisms that are polynomial-time in the size of the input.

That said, we note that a side effect is that our mechanisms are centralized whereas many of the classic mechanisms can be implemented in a decentralized way (e.g., Cut and Choose). In particular, it is unclear whether our mechanisms can be efficiently implemented via evaluation and cut queries. For example, it seems difficult to isolate the desired intervals of an agent using a finite number of such queries. This does not preclude, however, the existence of results that are analogous to our Theorems 3.1 and 4.3 via mechanisms that only rely on evaluation and cut queries.

Future work. The most prominent technical challenge is to generalize Mechanism 1. The first step would be a deterministic, truthful, proportional, and EF mechanism under the assumption that the agents have piecewise constant valuations, and the second step would be achieving the same result with respect to piecewise linear valuations. Unfortunately we cannot rule out the scenario where no such mechanisms exist; future work would have to resolve this issue.

For practical settings, allowing more expressiveness seems important. For example, consider a setting where the cake is commercial time on TV and the agents are advertisers. Even if an advertiser is allocated a desired time interval, the allocation is useless if the amount of time allocated is shorter than the advertiser's commercial. Hence it is natural to consider piecewise uniform valuations where each agent has a minimum interval length, and allocated intervals that are smaller than the minimum are worthless. Truthfulness issues aside, this setting is complicated by the fact that an EF allocation is not guaranteed to exist (e.g., the minimum with respect to all agents may be the entire cake), but one may consider *approximately* proportional and EF mechanisms [12]. This is the subject of ongoing research.

Acknowledgments

We thank Noga Alon for his wise advice regarding the tractable implementation of Step 2 of SubRoutine. We also thank Anna Bogomolnaia and Hervé Moulin for pointing our attention to the connection between our problem and the random assignment problem.

References

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

[2] A. Bogomolnaia and H. Moulin. A new solution to the random assignment problem. *Journal of Economic Theory*, 100:295–328, 2001.

[3] A. Bogomolnaia and H. Moulin. Random matching under dichotomous preferences. *Econometrica*, 72:257–279, 2004.

[4] S. J. Brams, M. A. Jones, and C. Klamler. Better ways to cut a cake. *Notices of the AMS*, 53(11):1314–1321, 2006.

[5] S. J. Brams, M. A. Jones, and C. Klamler. Proportional pie-cutting. *Int. Journal of Game Theory*, 36(3–4):353–367, 2008.

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

[7] C. Busch, M. S. Krishnamoorthy, and M. Magdon-Ismail. Hardness results for cake cutting. *Bulletin of the EATCS*, 86:85–106, 2005.

[8] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein. *Introduction to Algorithms.* MIT Press, 2nd edition, 2001.

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

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

[11] A. Katta and J. Sethuraman. A solution to the random assignment problem on the full preference domain. *Journal of Economic Theory*, 131:231–250, 2006.

[12] R. J. Lipton, E. Markakis, E. Mossel, and A. Saberi. On approximately fair allocations of indivisible goods. In *Proc. of 6th EC*, pages 125–131, 2004.

[13] N. Megiddo. A good algorithm for lexicographically optimal flows in multi-terminal networks. *Bulletin of the American Mathematical Society*, 83:407–409, 1979.

[14] E. Mossel and O. Tamuz. Truthful fair division. In *Proc. of 3rd SAGT*, 2010. To appear.

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

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

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

[18] W. Thomson. Children crying at birthday parties. Why? *Journal of Economic Theory*, 31:501–521, 2007.

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