ECAI 2024
U. Endriss et al. (Eds.)

doi:10.3233/FAIA240885

On Connected Strongly-Proportional Cake-Cutting

Zsuzsanna Jankó[a], Attila Joó[b], Erel Segal-Halevi[c,*] and Sheung Man Yuen[d]

[a]Corvinus University of Budapest, HUN-REN Centre for Economic and Regional Studies
[b]University of Hamburg
[c]Ariel University
[d]National University of Singapore

Abstract. We investigate the problem of fairly dividing a divisible heterogeneous resource, also known as a cake, among a set of agents who may have different entitlements. We characterize the existence of a connected strongly-proportional allocation—one in which every agent receives a contiguous piece worth strictly more than their proportional share. The characterization is supplemented with an algorithm that determines its existence using $O(n \cdot 2^n)$ queries. We devise a simpler characterization for agents with strictly positive valuations and with equal entitlements, and present an algorithm to determine the existence of such an allocation using $O(n^2)$ queries. We provide matching lower bounds in the number of queries for both algorithms. When a connected strongly-proportional allocation exists, we show that it can also be computed using a similar number of queries.

The full version is available at https://arxiv.org/abs/2312.15326.

1 Introduction

Consider a group of siblings who inherited a land estate, and would like to divide it fairly among themselves. The simplest procedure for attaining a fair division is to sell the land and divide the proceeds equally; this procedure guarantees each sibling a proportional share of the total land value.

But in some cases, it is possible to give each sibling a much better deal. As an example, suppose that the land estate contains one part that is fertile and arable, and one part that is barren but has potential for coal mining. This land is to be divided between two siblings, one of whom is a farmer and the other is a coal factory owner. If we give the former piece of land to the farmer and the latter piece of land to the coal factory owner, both siblings will feel that they receive more than half of the total land value. Our main question of interest is: when is such a superior allocation possible?

We study this question in the framework of *cake-cutting*. In this setting, there is a divisible resource called a *cake*, which can be cut into arbitrarily small pieces without losing its value. The cake is represented simply by an interval which can model a one-dimensional object, such as time. There are n agents, each of whom has a personal measure of value over the cake. The goal is to partition the cake into n pieces and allocate one piece per agent such that the agents feel that they receive a "fair share" according to some fairness notion.

A common fairness criterion—nowadays called *proportionality*—requires that each agent i receives a piece of cake that is worth, according to i's valuation, at least $1/n$ of the total cake value. In his seminal paper, Steinhaus [20] described an algorithm, developed by his students Banach and Knaster, that finds a proportional allocation; moreover, this allocation is *connected*—each agent receives a single contiguous part of the cake. This algorithm is now called the *last diminisher* algorithm.

But the guarantee of proportionality allows for the possibility that each agent receives a piece worth *exactly* $1/n$; when this is the case, there is little advantage in using a cake-cutting procedure over selling the land and giving $1/n$ to each partner. A stronger criterion, called *strong-proportionality* or *super-proportionality*, requires that each agent i receives a piece of cake worth *strictly more* than $1/n$ of the total cake value from i's perspective. This raises the question of when such a strongly-proportional allocation exists.

Obviously, a strongly-proportional allocation does not exist when all the agents' valuations are identical, since if any agent receives more than $1/n$ of the cake, then some other agent must receive less than $1/n$ of the cake. Interestingly, in all other cases, a strongly-proportional allocation exists. Even when *two* agents have non-identical valuations, there exists an allocation in which *all* n agents receive more than $1/n$ of the total cake value from their perspectives [8, 16]. Woodall [27] presented an algorithm for finding such a strongly-proportional allocation. Barbanel [2] generalized this algorithm to agents with unequal entitlements, and Jankó and Joó [13] presented a simple algorithm for this generalized problem and extended it to infinitely many agents.

The problem with all these algorithms is that, in contrast to the last diminisher algorithm for proportional cake-cutting, they do not guarantee a *connected* allocation. Connectivity is an important practical consideration when allocating cakes; for example, if the cake is the availability of a meeting room by time and needs to be allocated to different teams throughout the day, then a two-hour slot is easier for a team to utilize than six disjoint twenty-minute slots. Indeed, connectivity is the most commonly studied constraint in cake-cutting literature [24, 21, 23, 22, 11, 9], and relaxing this constraint may present each agent instead with a "countable union of crumbs" [21].

Thus, our main questions of interest are:

What are the necessary and sufficient conditions for the existence of a connected strongly-proportional cake allocation? What are the query complexities to determine these conditions?

1.1 Our Results

The cake to be allocated, modeled by a unit interval $[0, 1]$, is to be divided among n agents who may have different entitlements for the

* Corresponding Author. Email: erelsgl@gmail.com.

cake, with the entitlements summing to 1. Each agent receives an interval of the cake that is disjoint from the other agents' intervals. Each agent has a valuation function on the intervals of the cake that is non-negative, finitely additive, and continuous with respect to length. In this regard, the value of a single point is zero to every agent, and we can assume without loss of generality that agents receive *closed* intervals of the cake, and that any two agents' pieces can possibly intersect at the endpoints of their respective intervals.[1] In order to access agents' valuations in the algorithms, we allow algorithms to make eval and (right-)mark[2] queries of each agent as in the standard Robertson-Webb model [17]. More details of our model are provided in Section 2.

In Section 3, we consider *hungry* agents—those who have positive valuations for any part of the cake with positive length. For agents with equal entitlements, we show that a connected strongly-proportional allocation exists if and only if there are two agents with different r-marks for some $r \in \{1/n, 2/n, \ldots, (n-1)/n\}$, where an r-mark is a point that divides the cake into two such that the left part of the cake is worth r to that agent. This implies that the existence of such an allocation can be decided using $n(n-1)$ queries. The proof of sufficiency is constructive, so a connected strongly-proportional allocation can be computed using $O(n^2)$ queries if it exists. We also prove that any algorithm that decides whether a connected strongly-proportional allocation exists must make at least $n(n-1)/2$ queries, giving an asymptotically tight bound (within a factor of 2) of $\Theta(n^2)$. For agents with possibly unequal entitlements, we show that a lower bound number of queries to decide whether a connected strongly-proportional allocation exists is $n \cdot 2^{n-2}$. Together with a result from Section 4 later on the upper bound number of queries, this yields a tight bound of $\Theta(n \cdot 2^n)$ queries.

In Section 4, we consider agents who are not necessarily hungry. The characterization from Section 3 for hungry agents with equal entitlements does not work for non-hungry agents, which motivates us to find another characterization by considering permutations of agents. We show that a connected strongly-proportional allocation exists if and only if there exists a permutation of agents such that when the agents go in the order as prescribed by the permutation and make their rightmost marks worth their entitlements to each of them one after another, the mark made by the last agent does not reach the end of the cake. This result holds regardless of the agents' entitlements. While an algorithm to determine this condition requires $n \cdot n!$ queries, we show that this number can be reduced by a factor of $2^{\omega(n)}$ to $n \cdot 2^{n-1}$ via dynamic programming. We also prove a lower bound number of queries of $\Omega(n \cdot 2^n)$ to determine this condition, even for agents with equal entitlements. Therefore, for agents who are not necessarily hungry, we also obtain a tight bound of $\Theta(n \cdot 2^n)$, whether the entitlements are equal or not. A connected strongly-proportional allocation can be computed using $O(n \cdot 2^n)$ queries if it exists.

Table 1 summarizes of our results from Sections 3 and 4. All omitted proofs can be found in the full version of our paper [14].

1.2 Further Related Work

A weaker fairness notion of *proportionality* is well-studied in cake-cutting literature. It is known that a connected proportional allocation always exists for agents with equal entitlements and such an allocation can be computed using $\Theta(n \log n)$ queries [20, 10, 26]. Cseh and Fleiner [7] presented an algorithm that finds a possibly non-connected proportional allocation for agents with general entitlements—in particular, their algorithm uses a finite but *unbounded* number of queries when agents have irrational entitlements. In contrast, we show that a connected *strongly-proportional* allocation may not exist, and such an allocation can be computed (if it exists) using $\Theta(n \cdot 2^n)$ queries. A number of works studied the number of *cuts* required for a proportional allocation, rather than the number of queries [18, 6].

A parallel line of work studied a stronger fairness notion of *super envy-freeness*: it requires, in addition to strong-proportionality, that each agent values the piece of every other agent at strictly less than $1/n$ the total cake value [3, 25, 5].

Table 1. Number of queries required to decide the existence of a connected strongly-proportional allocation of a cake for n agents, and to compute one if it exists.

	hungry agents	**general agents**
equal entitlements	$\Theta(n^2)$ (Thm 3.5)	$\Theta(n \cdot 2^n)$ (Thm 4.5)
possibly unequal entitlements	$\Theta(n \cdot 2^n)$ (Thm 3.7)	$\Theta(n \cdot 2^n)$ (Thm 4.5)

2 Preliminaries

Let the cake be denoted by $C = [0, 1]$. The cake is to be allocated to a set of agents denoted by $[n] := \{1, \ldots, n\}$. A *piece of cake* is a finite union of closed intervals of the cake. An *allocation* of C is a partition of C into n pairwise-disjoint[3] pieces of cake $(X_1, \ldots, X_n)$ such that $C = X_1 \sqcup \cdots \sqcup X_n$; X_i is the piece allocated to agent i. An allocation is *connected* if X_i is a single interval for each $i \in [n]$.

The preference of each agent i is represented by a *valuation function* V_i such that $V_i(X)$ is the value of the piece $X \subseteq C$ to agent i. Each valuation function V_i is defined on the algebra over C generated by all intervals of C, and is non-negative (i.e., $V_i(X) \geq 0$ for all $X \subseteq C$ in the algebra), finitely additive (i.e., $V_i(X \cup Y) = V_i(X) + V_i(Y)$ for all disjoint $X, Y \subseteq C$ in the algebra), and normalized to one (i.e., $V_i(C) = 1$). We assume that $F_i(x) := V_i([0, x])$ is a continuous function on C, and hence $V_i(\{x\}) = 0$ for all $x \in C$. Therefore, F_i is a non-decreasing function on C with $F_i(0) = 0$, $F_i(1) = 1$, and $V_i([x, y]) = F_i(y) - F_i(x)$. An agent i is *hungry* if $V_i(X) > 0$ for all intervals $X \subseteq C$ with positive length; this is equivalent to the condition that F_i is strictly increasing.

Each agent i has an *entitlement* $w_i > 0$ of the cake such that $\sum_{i \in [n]} w_i = 1$. Let $\mathbf{w}$ denote $(w_1, \ldots, w_n)$. We say that agents have *equal entitlements* if $w_i = 1/n$ for all $i \in [n]$. For each subset $N \subseteq [n]$ of agents, define $w_N = \sum_{i \in N} w_i$. Note that $w_\varnothing = 0$ and $w_{[n]} = 1$. We say that agents have *generic entitlements* if $w_N \neq w_{N'}$ for all distinct $N, N' \subseteq [n]$.

A *(cake-cutting) instance* consists of the set of agents, their valuation functions $(V_i)_{i \in [n]}$, and their entitlements $\mathbf{w}$.

Given an instance, an allocation $(X_1, \ldots, X_n)$ is *proportional* (resp. *strongly-proportional*) if $V_i(X_i) \geq w_i$ (resp. $V_i(X_i) > w_i$) for all $i \in [n]$. For agents with equal entitlements, a proportional (resp. strongly-proportional) allocation requires every agent to receive a piece of cake with value at least (resp. greater than) $1/n$.

Algorithms can make eval and mark queries of each agent in the Robertson-Webb model. More specifically, for each agent $i \in [n]$,

[1] This is often assumed in cake-cutting literature; see e.g. Procaccia [15].

[2] We choose *right*-mark instead of the usual *left*-mark for convenience. Our algorithms still work if only left-mark queries are available (together with eval). See the full version of our paper [14] for a more detailed explanation.

[3] As mentioned in Section 1.1, two pieces of cake are also considered disjoint if their intersection is a subset of the endpoints of their respective intervals.

value $r \in [0,1]$, and points $x, y \in C$ with $x \leq y$, $\text{EVAL}_i(x,y)$ returns $V_i([x,y])$, and $\text{MARK}_i(x,r)$ returns the *rightmost* (largest) point $z \in C$ such that $V_i([x,z]) = r$ (such a point exists due to the continuity of the valuations); if $V_i([x,1]) < r$, then $\text{MARK}_i(x,r)$ returns ∞.

For $i \in [n]$ and $r \in [0,1]$, a point $x \in C$ is an *r-mark* of agent i if $V_i([0,x]) = r$. While the point returned by $\text{MARK}_i(0,r)$ is an r-mark of agent i, the converse is not true since $\text{MARK}_i(0,r)$ only returns the *rightmost* r-mark of agent i. However, when agent i is *hungry*, then the r-mark is unique, and the two notions coincide. Let $\mathcal{T}$ denote the subset $\{1/n, 2/n, \ldots, (n-1)/n\}$ of C—we shall consider r-marks for $r \in \mathcal{T}$ in Section 3.1.

3 Hungry Agents

We begin with the simpler case where all agents are hungry. We first state a result which finds a connected strongly-proportional allocation of a cake for hungry agents using a small number of queries when given a connected *proportional* allocation in which one agent has a strongly-proportional piece. The proof proceeds by slightly moving the boundary between two adjacent agents' pieces such that an agent j who received exactly w_j eventually gets a slightly larger piece.

Lemma 3.1. *Let an instance with n hungry agents be given. Suppose that we are given a connected proportional allocation $(X_1, \ldots, X_n)$ such that $V_i(X_i) > w_i$ for some $i \in [n]$. Then, there exists a connected strongly-proportional allocation, and such an allocation can be computed using $O(n)$ queries.*

Proof. First, we find the values of $V_j(X_j)$ for all $j \in [n]$. If $V_j(X_j) > w_j$ for all $j \in [n]$, then we are done. Otherwise, there exist two distinct agents $i, j \in [n]$ with neighboring pieces such that $V_i(X_i) > w_i$ and $V_j(X_j) = w_j$. By slightly moving the boundary between X_i and X_j, we can get a new allocation in which agents i and j each receives a piece worth more than w_i and w_j respectively. To formally describe the process of moving the boundary, we consider two complementary cases.

Case 1: X_i is to the left of X_j. Denote $X_i = [z_1, z_2]$ and $X_j = [z_2, z_3]$. Let $y = \text{MARK}_i(z_1, w_i)$; note that $y \in (z_1, z_2)$ since $V_i(X_i) > w_i$. Let y^* be the midpoint of y and z_2. Adjust the two agents' pieces such that agent i now receives $[z_1, y^*]$ and agent j now receives $[y^*, z_3]$; see Figure 1 for an illustration.

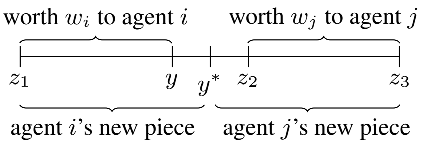

Figure 1. Agent i's and j's new pieces in the proof of Lemma 3.1.

Since $[z_1, y^*] \supsetneq [z_1, y]$ and the latter is worth w_i to hungry agent i, the new piece, $[z_1, y^*]$, is worth more than w_i to agent i. Likewise, since $[y^*, z_3] \supsetneq [z_2, z_3]$ and the latter is worth w_j to hungry agent j, the new piece, $[y^*, z_3]$, is worth more than w_j to agent j.

Case 2: X_i is to the right of X_j. Denote $X_j = [z_1, z_2]$ and $X_i = [z_2, z_3]$. Let $y = \text{MARK}_i(z_2, V_i(X_i) - w_i)$; note that $y \in (z_2, z_3)$ since $V_i(X_i) > w_i$. Let y^* be the midpoint of z_2 and y. Adjust the two agents' pieces such that agent j now receives $[z_1, y^*]$ and agent i now receives $[y^*, z_3]$.

Since $[z_1, y^*] \supsetneq [z_1, z_2]$ and the latter is worth w_j to hungry agent j, the new piece, $[z_1, y^*]$, is worth more than w_j to agent j. Likewise, since $[y^*, z_3] \supsetneq [y, z_3]$ and the latter is worth w_i to hungry agent i (due to additivity, we have $V_i(y, z_3) = V_i(z_2, z_3) - V_i(z_2, y) = w_i$), the new piece, $[y^*, z_3]$, is worth more than w_i to agent i.

In both Case 1 and Case 2, only agent i's and j's pieces change; all of the other agents' pieces do not change. All in all, one additional agent j receives more than w_j of the cake. Proceeding this way at most $n-1$ times yields a connected strongly-proportional allocation.

Finding the values of all $V_j(X_j)$ at the beginning requires n queries, while the adjustment of the boundaries between two agents' pieces requires a constant number of queries, so the total number of queries is in $O(n)$. □

We present the results separately for agents with equal entitlements and agents with possibly unequal entitlements. For n hungry agents with equal entitlements, we state in Section 3.1 a simple necessary and sufficient condition for the existence of a connected strongly-proportional allocation. We provide an asymptotically tight bound of $\Theta(n^2)$ for the number of queries needed by an algorithm to determine the existence of such an allocation, as well as to compute one such allocation if it exists. For agents with possibly unequal entitlements, we show in Section 3.2 that a lower bound number of queries needed to decide the existence of a connected strongly-proportional allocation is in $\Omega(n \cdot 2^n)$.

3.1 Equal Entitlements

Recall that $\mathcal{T} = \{1/n, 2/n, \ldots, (n-1)/n\}$. Our condition uses a particular set of r-marks: those with $r \in \mathcal{T}$.

Theorem 3.2. *Let an instance with n hungry agents with equal entitlements be given. Then, a connected strongly-proportional allocation exists if and only if there exist two distinct agents $i, j \in [n]$ and $r \in \mathcal{T}$ such that the r-mark of agent i is different from the r-mark of agent j.*

Proof. Since the agents are hungry, there is exactly one r-mark of agent i for each $r \in [0,1]$ and $i \in [n]$.

($\Rightarrow$) We prove the contraposition. Suppose that for each $r \in \mathcal{T}$, every agent has the same r-mark. Every agent also has the same 0-mark of 0 and the same 1-mark of 1. For each $t \in \{0, \ldots, n\}$, denote the common t/n-mark by z_t.

Consider now any connected allocation, which is represented by $n-1$ cuts on the cake. For each $t \in [n-1]$, denote the t-th cut from the left by x_t; also denote $x_0 = 0$ and $x_n = 1$. Each agent receives a piece $[x_{t-1}, x_t]$ for some $t \in [n]$, and every such piece is allocated to some agent.

Since $x_0 = z_0$ and $x_n = z_n$, there must be some $t \in [n]$ for which $x_{t-1} \geq z_{t-1}$ and $x_t \leq z_t$. This means that the piece $[x_{t-1}, x_t]$ is contained in the interval $[z_{t-1}, z_t]$. Let i denote the agent who receives the piece $[x_{t-1}, x_t]$. Then, agent i's value for her piece is

$$\begin{aligned} V_i([x_{t-1}, x_t]) &\leq V_i([z_{t-1}, z_t]) \\ &= V_i([0, z_t]) - V_i([0, z_{t-1}]) \\ &= t/n - (t-1)/n = 1/n, \end{aligned}$$

so the allocation is not strongly-proportional. This holds for any connected allocation; therefore, no connected strongly-proportional allocation exists.

($\Leftarrow$) Suppose that there exist two distinct agents $i, j \in [n]$ and $r \in \mathcal{T}$ such that the r-mark of agent i is different from the r-mark of agent j. We shall construct a connected strongly-proportional allocation by first constructing a connected *proportional* allocation such that at least one agent receives a piece with value more than $1/n$, then use Lemma 3.1 to construct a strongly-proportional one.

Let $t \in [n-1]$ be the integer such that $r = t/n$. Let i_L be an agent with the leftmost (smallest) r-mark among all the agents, and i_R be an agent with the rightmost (largest) r-mark among all the agents (if there are multiple agents with the same leftmost or rightmost r-mark, we can choose an agent arbitrarily in each case). Denote the leftmost r-mark by z_L and the rightmost r-mark by z_R. Note that $z_L < z_R$, since there are agents with different r-marks.

Since there are n agents, there are n r-marks (possibly some of them are equal) in the interval $[z_L, z_R]$. Let $x \in [z_L, z_R]$ be the t-th r-mark from the left. Then, there exists a partition of the agents into two subsets N_1 and N_2 such that

- $|N_1| = t$, and the r-mark of all agents in N_1 is at most x, and
- $|N_2| = n - t$, and the r-mark of all agents in N_2 is at least x.

Every agent in N_1 values $[0, x]$ at least r, and every agent in N_2 values $[x, 1]$ at least $1 - r$; see Figure 2 for an illustration.

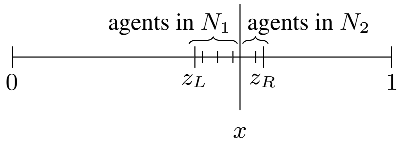

Figure 2. The r-marks of all the agents in the proof of Theorem 3.2. The point x is at one of the r-marks and divides agents into N_1 and N_2.

Next, we consider any connected proportional cake-cutting algorithm as a black box (e.g., last diminisher). We apply the algorithm on $[0, x]$ and N_1 such that every agent in N_1 receives a connected piece with value at least $1/t$ of her value of $[0, x]$, and apply the algorithm on $[x, 1]$ and N_2 such that every agent in N_2 receives a connected piece with value at least $1/(n-t)$ of her value of $[x, 1]$. We show that this allocation (of $C = [0, 1]$) is proportional. For an agent in N_1, since she values $[0, x]$ at least $r = t/n$, the piece she receives has value at least $(1/t)r = 1/n$. Likewise, for an agent in N_2, since she values $[x, 1]$ at least $1 - r = (n-t)/n$, the piece she receives has value at least $(1/(n-t))(1-r) = 1/n$.

Now, we show that agent i_L or i_R (or both) receives a piece with value strictly more than $1/n$. If $x = z_R$, then we claim that agent i_L receives such a piece. Since the r-mark of agent i_L is at $z_L < x$, we have $i_L \in N_1$. Since agent i_L is hungry, the piece $[0, x]$ is worth more than r to her, and so the piece she receives has value more than $(1/t)r = 1/n$. Otherwise, $x < z_R$, and a similar argument shows that agent i_R receives such a piece.

Having established a connected proportional allocation in which at least one agent receives more than $1/n$, we apply Lemma 3.1 to obtain a connected *strongly-proportional* allocation. □

It is interesting to compare the condition in Theorem 3.2 with the one for non-connected allocations. In both cases, a disagreement between *two* agents is sufficient for allocating *all* n agents more than their fair share. However, in the non-connected case, the disagreement can be in an r-mark for any $r \in (0, 1)$ (see the discussion in Section 1), whereas in the connected case, the disagreement should be in an r-mark for some $r \in \mathcal{T}$; the r-marks for other values of r are completely irrelevant.

It is clear from Theorem 3.2 that we can decide whether a connected strongly-proportional allocation exists for hungry agents with equal entitlements by checking the t/n-marks of all of the n agents for all $t \in [n-1]$. This is described in Algorithm 1. The number of queries used in the algorithm is at most $n(n-1)$.

Algorithm 1 Determining the existence of a connected strongly-proportional allocation for n hungry agents with equal entitlements.

for $t = 1, \ldots, n-1$ **do**
  $z_t \leftarrow \textsc{Mark}_1(0, t/n)$ ▷ agent 1's t/n-mark
  for $i = 2, \ldots, n$ **do**
    if $\textsc{Mark}_i(0, t/n) \neq z_t$ **then return** true
  end for
end for
return false

Theorem 3.3. *Algorithm 1 decides whether a connected strongly-proportional allocation exists for n hungry agents with equal entitlements using at most $n(n-1)$ queries.*

Next, we show an asymptotically tight lower bound for the number of queries required to decide the existence of such an allocation for hungry agents. The idea behind the proof is that we must check the t/n-marks of all the agents and all $t \in [n-1]$; otherwise, we can craft two instances—one with the t/n-marks coinciding, and the other with some t/n-marks not coinciding—that are consistent with the information obtained by the algorithm and yet give opposite results. Doing this check requires at least $n(n-1)/2$ queries, as each query provides information on at most two points.

Theorem 3.4. *Any algorithm that decides whether a connected strongly-proportional allocation exists for n hungry agents with equal entitlements requires at least $n(n-1)/2$ queries.*

Proof. Suppose by way of contradiction that some algorithm decides the existence of a connected strongly-proportional allocation for n hungry agents with equal entitlements using fewer than $n(n-1)/2$ queries. We assume that for all $i \in [n]$, $r \in [0, 1]$ and $x \in C$, $\textsc{Eval}_i(0, x)$ returns the value x and $\textsc{Mark}_i(0, r)$ returns the point r. We make the following adjustments to the algorithm: whenever the algorithm makes an $\textsc{Eval}_i(x, y)$ query, it is instead given the answers to $\textsc{Mark}_i(0, x) = x$ and $\textsc{Mark}_i(0, y) = y$, and whenever the algorithm makes a $\textsc{Mark}_i(x, r)$ query, it is instead given the answers to $\textsc{Mark}_i(0, x) = x$ and $\textsc{Mark}_i(0, x + r) = x + r$.[4] This means that every query made by the algorithm provides the algorithm only with information on at most *two* r-marks of some agent and no other information that cannot be deduced from these r-marks. Note that the algorithm can still deduce the values of $\textsc{Eval}_i(x, y)$ and $\textsc{Mark}_i(x, r)$ by taking the difference between the two answers given, which means that the information provided to the algorithm after the adjustment is a superset of the information provided to the algorithm before the adjustment.

The answers given to the algorithm are consistent with the instance where every agent's valuation is uniformly distributed over the cake—in which case there is no connected strongly-proportional allocation of the cake by Theorem 3.2—and so the algorithm should output "false". However, we shall now show that the information provided to the algorithm is also consistent with an instance with a connected strongly-proportional allocation. This means that the algorithm is not able to differentiate between the two, resulting in a contradiction.

Since fewer than $n(n-1)/2$ queries were made by the algorithm, fewer than $n(n-1)$ r-marks (for $r \in (0, 1)$) of all the agents are

[4] Assuming $x + r \leq 1$; otherwise, $\textsc{Mark}_i(0, x + r) = \infty$.

known. In particular, there exists an agent $i \in [n]$ such that fewer than $n-1$ r-marks of agent i are known, and hence there exists $t \in [n-1]$ such that the t/n-mark of agent i is not known. We now modify agent i's valuation function slightly from the uniform distribution. Let $\epsilon \in (0, 1/n)$ be a number such that every known r-mark of agent i is of distance more than ϵ from t/n. Let the t/n-mark of agent i to be at $t/n+\epsilon$. Construct agent i's valuation function such that its distribution between all known r-marks of agent i (including the new t/n-mark) is uniform within the respective intervals—note that this construction is valid and unique since these known r-marks are strictly increasing in r. Let the other agents' valuation functions be uniformly distributed on the whole cake. Then, agent i's t/n-mark is different from every other agents' t/n-mark. By Theorem 3.2, this instance admits a connected strongly-proportional allocation of the cake, forming the desired contradiction. □

Theorems 3.3 and 3.4 show that the number of queries required to determine the existence of a connected strongly-proportional allocation for n hungry agents with equal entitlements is in $\Theta(n^2)$. The same can be said for *computing* such an allocation—we can modify Algorithm 1 using the details in the proof of Theorem 3.2 to output a connected strongly-proportional allocation of the cake instead, if such an allocation exists.

Theorem 3.5. *The number of queries required to decide the existence of a connected strongly-proportional allocation for n hungry agents with equal entitlements, or to compute such an allocation if it exists, is in $\Theta(n^2)$.*

3.2 Possibly Unequal Entitlements

We now consider hungry agents who may not necessarily have equal entitlements. Since the entitlement of a subset of agents may not be a multiple of $1/n$, we cannot use the condition in Theorem 3.2 which uses r-marks for $r \in \mathcal{T}$. This requires us to devise a more general condition to determine the existence of a connected strongly-proportional allocation, which can be checked using $O(n \cdot 2^n)$ queries. Since the condition also works for non-hungry agents, we defer the discussion to Section 4.1 (see Theorems 4.2 and 4.3).

We now show an asymptotically-tight *lower bound* for the case when agents may have unequal entitlements. We show an even stronger result: for every vector of *generic entitlements*, the number of queries required to decide the existence of a connected strongly-proportional allocation is in $\Omega(n \cdot 2^n)$. The proof uses an adversarial argument similar to the one in Theorem 3.4.

Theorem 3.6. *Let $\mathbf{w}$ be any vector of generic entitlements. Then, any algorithm that decides whether a connected strongly-proportional allocation exists for n hungry agents with entitlements $\mathbf{w}$ requires at least $n \cdot 2^{n-2}$ queries.*

Proof. Since the entitlements are generic, we can arrange the 2^n different subsets of agents in strictly increasing order of their entitlements, i.e., we label the subsets of $[n]$ as $N_1, \ldots, N_{2^n}$ such that $w_{N_1} < \cdots < w_{N_{2^n}}$. Note that $N_1 = \varnothing$ and $N_{2^n} = [n]$, giving $w_{N_1} = 0$ and $w_{N_{2^n}} = 1$.

Let $d = \min_{k=1}^{2^n-1}(w_{N_{k+1}} - w_{N_k})$ be the smallest gap between entitlements of different agent subsets. For each $k \in \{2, \ldots, 2^n-1\}$, define $I_k = [w_{N_k},\ w_{N_k} + d/2]$. Note that, by the choice of d, all the I_k are pairwise disjoint.

Suppose by way of contradiction that some algorithm decides the existence of a connected strongly-proportional allocation for n hungry agents with generic entitlements using fewer than $n \cdot 2^{n-2}$ queries. We follow the construction in the proof of Theorem 3.4 where we modify the algorithm such that every query returns information on at most *two* r-marks of some agent, and these information are consistent with the instance where every agent's valuation is uniformly distributed over the cake. Therefore, the algorithm should output "false". We shall now show that the information provided to the algorithm is also consistent with an instance with a connected strongly-proportional allocation. This means that the algorithm is not able to differentiate between the two, resulting in a contradiction.

Since fewer than $n \cdot 2^{n-2}$ queries were made by the algorithm, there exists an agent $i \in [n]$ such that at most $2^{n-2}-1$ queries about the r-marks of agent i (for $r \in (0,1)$) are made. Since each query returns information on at most *two* r-marks, at most $2^{n-1}-2$ r-marks of agent i are known. There are $2^{n-1}-1$ non-empty subsets N_k of $[n]$ that do not contain agent i, so there exists $k \in \{2, \ldots, 2^n-1\}$ such that $i \notin N_k$ and no known r-mark of agent i is in the interval I_k. Let $w = w_{N_k}$. Let the w-mark of agent i be at $w + d/4$. Construct agent i's valuation function such that its distribution between all known r-marks of agent i (including the new w-mark) is uniform within the respective intervals—note that this construction is valid and unique since these known r-marks are strictly increasing in r. Let the other agents' valuation functions be uniformly distributed on the whole cake.

We show that a connected strongly-proportional allocation exists. The leftmost pieces are allocated to agents in N_k in any arbitrary order, where every agent $j \in N_k$ receives a piece of length w_j. Agent i receives the piece $[w, w + w_i]$. Finally, the remaining cake is allocated to the remaining agents such that every agent j receives a piece of length w_j. Note that every agent $j \in [n] \setminus \{i\}$ receives a piece worth exactly w_j, since their valuation functions are uniform. The value of $[w + d/4, w + w_i]$ is w_i to agent i, so agent i's piece $[w, w + w_i] \supsetneq [w + d/4, w + w_i]$ is worth more than w_i to hungry agent i. Therefore, the allocation is proportional (and clearly connected) with agent i receiving a piece strictly greater than w_i. By Lemma 3.1, a connected strongly-proportional allocation of the cake exists, forming the desired contradiction. □

Using the results from Theorem 3.6 and from Theorem 4.3 later, we get a tight bound for hungry agents with possibly unequal entitlements.

Theorem 3.7. *The number of queries required to decide the existence of a connected strongly-proportional allocation for n hungry agents, or to compute such an allocation if it exists, is in $\Theta(n \cdot 2^n)$.*

The lower bound in Theorem 3.6 is derived from the number of different values of w_{N_k}. In particular, a lower bound number of queries is

$$\frac{1}{2} \sum_{i=1}^{n} |\{w_N : \varnothing \neq N \subseteq [n], i \notin N\}|. \tag{1}$$

For *generic* entitlements, each term in the sum equals $2^{n-1}-1$, so we get roughly the lower bound of $n \cdot 2^{n-2}$ in Theorem 3.6. In contrast, for *equal* entitlements, each term in the sum equals $n-1$, so we get the lower bound of $n(n-1)/2$ in Theorem 3.4.

For entitlements that are neither generic nor equal, the resulting lower bound is between these two extremes. It is an interesting open question to find an algorithm with a query complexity matching the lower bound in (1) in these intermediate cases. The main difficulty in extending our algorithm for equal entitlements (Algorithm 1) to unequal entitlements is due to the step in Theorem 3.2 where we used a black-box algorithm for *proportional* cake-cutting (such as last diminisher) to divide a part of the cake among the agents in N_1 and the other part among the agents in N_2. Such a black box algorithm does not exist for unequal entitlements, since a connected proportional allocation might not even exist for unequal entitlements in the first place.

4 General Agents

We now consider the general case where agents need not be hungry. Recall that the condition we developed in Theorem 3.2 involves checking for the coincidence of r-marks of all the agents for $r \in \mathcal{T}$. However, there are some difficulties in generalizing the condition for non-hungry agents, even for equal entitlements. The proof of Theorem 3.2 relies crucially on the fact that an r-mark of an agent is unique, which may not be true for non-hungry agents. In fact, the set of r-marks of agent i is a non-empty closed *interval* (though possibly the singleton set $[x, x] = \{x\}$). There is a natural generalization of the condition for two agents which we state as Proposition 4.1 below.

Proposition 4.1. *Let an instance with two agents with equal entitlements be given. Then, a connected strongly-proportional allocation exists if and only if the intervals of* $1/2$*-marks of the two agents are disjoint.*[5]

We provide the proof of Proposition 4.1, as well as a discussion of why the condition cannot be generalized to three or more agents, in the full version of our paper [14]. This inspires us to find another condition that characterizes the existence of a connected strongly-proportional allocation.

In Section 4.1, we generalize the condition from Theorem 3.2 for n non-hungry agents, regardless of whether they have equal entitlements or not. We show that this condition can be checked by an algorithm using $O(n \cdot 2^n)$ queries. Now, the result in Theorem 3.6 says that the lower bound number of queries needed for an algorithm to determine the existence of a connected strongly-proportional allocation for n hungry agents with generic entitlements is $\Omega(n \cdot 2^n)$—we show in Section 4.2 that this lower bound also applies to (not necessarily hungry) agents with *equal entitlements*.

4.1 Upper Bound

Our condition requires agents to mark pieces of cake one after another in a certain order. We explain this operation more precisely. Let $\sigma : [n] \to [n]$ be a permutation of agents, and let $x \in C$ and $r_1, \ldots, r_n \in [0, 1]$. The agents proceed in the order $\sigma(1), \ldots, \sigma(n)$. Agent $\sigma(1)$ starts first and makes a mark at $x_1 = \textsc{Mark}_{\sigma(1)}(x, r_{\sigma(1)})$, the rightmost point such that $[x, x_1]$ is worth $r_{\sigma(1)}$ to her. Then, agent $\sigma(2)$ continues from x_1, and makes a mark at $x_2 = \textsc{Mark}_{\sigma(2)}(x_1, r_{\sigma(2)})$, the rightmost point such that $[x_1, x_2]$ is worth $r_{\sigma(2)}$ to her. Each agent $\sigma(i)$ repeats the same process of making a mark at $x_i = \textsc{Mark}_{\sigma(i)}(x_{i-1}, r_{\sigma(i)})$ such that $[x_{i-1}, x_i]$ is the largest possible piece worth $r_{\sigma(i)}$ to her. We shall overload the definition of Mark and define[6] $\textsc{Mark}_\sigma(x, \mathbf{r})$ as the point x_n resulting from this sequential marking process, where $\mathbf{r} = (r_1, \ldots, r_n)$. If $[x_{i-1}, 1]$ is worth less than $r_{\sigma(i)}$ to agent $\sigma(i)$ at any point, then $\textsc{Mark}_\sigma(x, \mathbf{r})$ is defined as ∞. This operation is described in Algorithm 2. Note that each $\textsc{Mark}_\sigma(x, \mathbf{r})$ operation requires at most n ($\textsc{Mark}_i$) queries.

[5] Unlike for pieces of cake where "disjoint" means *finite* intersection, we revert to the standard definition of "disjoint" to mean *empty* intersection for intervals involving r-marks.

[6] The subscript of Mark here is a permutation σ, not an agent number.

Algorithm 2 Computing $\textsc{Mark}_\sigma(x, \mathbf{r})$ for n agents.

$x_0 \leftarrow x$
for $i = 1, \ldots, n$ **do**
  $x_i \leftarrow \textsc{Mark}_{\sigma(i)}(x_{i-1}, r_{\sigma(i)})$
  if $x_i = \infty$ **then return** ∞
end for
return x_n

Our necessary and sufficient condition for n (possibly non-hungry) agents requires us to check whether the point $\textsc{Mark}_\sigma(0, \mathbf{w})$ is less than 1 for some permutation σ. The point $\textsc{Mark}_\sigma(0, \mathbf{w})$ is determined when agents go in the order as prescribed by σ and make their rightmost marks worth their entitlements to each of them one after another. The idea behind the proof is that starting from the agent who receives the rightmost piece in σ and going leftwards, each agent is able to move the boundaries of her piece such that she receives a small piece of cake with positive value ϵ from the right and gives away a small piece of cake with value $\epsilon/2$ to the agent on the left, thereby increasing the value of her piece by a positive value $\epsilon/2$.

Theorem 4.2. *Let an instance with* n *agents be given. Then, a connected strongly-proportional allocation exists if and only if there exists a permutation* $\sigma : [n] \to [n]$ *such that* $\textsc{Mark}_\sigma(0, \mathbf{w}) < 1$.

The condition in Theorem 4.2 reduces to the condition in Theorem 3.2 for hungry agents with equal entitlements, i.e., when $\mathbf{w} = (1/n, \ldots, 1/n)$. In particular, when every agent has the same r-mark for each $r \in \mathcal{T}$, then each of the n marks made in the $\textsc{Mark}_\sigma(0, \mathbf{w})$ operation coincides at some $x_i \in \mathcal{T} \cup \{1\}$ for every permutation, and so $\textsc{Mark}_\sigma(0, \mathbf{w}) = 1$ for all σ. This corresponds to the case where no connected strongly-proportional allocation exists.

We can determine whether the condition in Theorem 4.2 holds by checking all permutation σ to see whether the point $\textsc{Mark}_\sigma(0, \mathbf{w})$ is less than 1 for some σ. Since there are $n!$ possible permutations of $[n]$ and each $\textsc{Mark}_\sigma$ operation requires at most n queries, the total number of queries required in the algorithm is at most $n \cdot n!$.

However, we can reduce the number of queries to $n \cdot 2^{n-1}$ by dynamic programming. Our approach is similar to the method used in Aumann et al. [1]—in their work, they iteratively find a value k such that there exists a connected allocation where every agent receives *at least* k, while here we require every agent i to receive a connected piece with value *strictly more* than w_i.

We now describe our algorithm. For every subset $N \subseteq [n]$, our algorithm caches the *best* mark b_N obtained by the subset of agents. The best mark b_N is the leftmost point possible over all permutations of the agents in N when the agents go in the order as prescribed by the permutation and make their rightmost marks worth their entitlements to each of them one after another. The algorithm aims to compute this point for every N.

The best mark for the empty set of agents is initialized as $b_\varnothing = 0$. Thereafter, for every $k \in [n]$, we assume that the best mark for every subset of $k - 1$ agents is calculated earlier and cached. We now need to find b_N for every subset $N \subseteq [n]$ with k agents. The last agent to make the best mark for N could be any of the agents $i \in N$. Therefore, for each $i \in N$, we retrieve the best mark for $N \setminus \{i\}$, which is $b_{N \setminus \{i\}}$ and has been cached earlier, and let agent i make the rightmost mark such that the cake starting from $b_{N \setminus \{i\}}$ is worth w_i to her. By iterating through all $i \in N$, we find the leftmost such point and cache this point as b_N. When $k = n$, we obtain $b_{[n]}$, which is the best $\textsc{Mark}_\sigma(0, \mathbf{w})$ over all permutations σ. Therefore, the algorithm returns "true" if $b_{[n]} < 1$, and "false" otherwise. This implementation reduces the number of queries by a factor of $2^{\omega(n)}$.

Agent 1	0	$a_1/(n-2)$		0	$a_1/(n-2)$	0	$1-a_1$	0
$\vdots$	$\vdots$	$\vdots$	$\cdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
Agent $n-1$	0	$a_{n-1}/(n-2)$	(total: $n-2$ identical copies)	0	$a_{n-1}/(n-2)$	0	$1-a_{n-1}$	0
Agent n	$1/n$	0		$1/n$	0	$1/n$	0	$1/n$

Figure 3. Construction of the cake used in the proof of Theorem 4.4.

This algorithm is described in Algorithm 3. The correctness of the algorithm relies on the statement in Theorem 4.2 and the fact that $b_{[n]}$ in the algorithm is less than 1 if and only if there exists a permutation $\sigma : [n] \to [n]$ such that $\text{MARK}_\sigma(0, \mathbf{w}) < 1$. For each $k \in [n]$, there are $\binom{n}{k}$ subsets N with cardinality k, and for each N, each of the $|N| = k$ agents makes a mark query—this means that $k\binom{n}{k}$ queries are made. Hence, the total number of queries is $\sum_{k=1}^{n} k\binom{n}{k} = n \cdot 2^{n-1}$ by a combinatorial identity.

Algorithm 3 Determining the existence of a connected strongly-proportional allocation for n agents.

$b_\varnothing \leftarrow 0$
for $k = 1, \ldots, n$ **do**
  for each subset $N \subseteq [n]$ with $|N| = k$ **do**
    $b_N \leftarrow \infty$
    for each agent $i \in N$ **do**
      $y \leftarrow \text{MARK}_i(b_{N\setminus\{i\}}, w_i)$
      if $y < b_N$ **then** $b_N \leftarrow y$ ▷ this finds the "best" b_N
    end for
  end for
end for
if $b_{[n]} < 1$ **then return** true **else return** false

Theorem 4.3. *Algorithm 3 decides whether a connected strongly-proportional allocation exists for n agents using at most $n \cdot 2^{n-1}$ queries.*

4.2 Lower Bound

Theorem 3.6 provides a lower bound for hungry agents with unequal entitlements; we shall now prove a similar lower bound for general agents with equal entitlements.

At a high level, the technique used is similar to that in the proofs of Theorems 3.4 and 3.6: we use an adversarial argument where we construct an instance with agents having uniform valuations on the cake such that no strongly-proportional allocation exists, but tweak the valuations slightly depending on the queries made. However, the details from the proof of Theorem 3.4 cannot be used directly since the existence of a connected strongly-proportional allocation is not solely dependent on the r-marks for $r \in \mathcal{T}$ for non-hungry agents [14], and the details from the proof of Theorem 3.6 cannot be used directly since Theorem 3.6 requires the entitlements to be generic.

Instead, we construct the following instance with $n \geq 3$ agents. The cake is divided into $2n-1$ parts. The odd parts (i.e., the 1st, 3rd, ..., $(2n-1)$-th parts) are non-valuable to agents 1 to $n-1$, and worth $1/n$ each to agent n. The even parts (i.e., the 2nd, 4th, ..., $(2n-2)$-th parts) are valuable to agents 1 to $n-1$, and non-valuable to agent n. For $i \in [n-1]$, agent i's first $n-2$ valuable parts (i.e., the 2nd, 4th, ..., $(2n-4)$-th parts) are worth $a_i/(n-2)$ each to agent i for some carefully selected a_i, and the last valuable part (i.e., the $(2n-2)$-th part) is worth $1-a_i$ to agent i. See Figure 3 for an illustration.

Consider a connected strongly-proportional allocation with equal entitlements. Agent n's piece has to include pieces from at least two consecutive odd parts in order for her value to be greater than $1/n$. By a clever choice of a_i for $i \in [n-1]$, we force these two odd parts to be the *rightmost* odd parts. This leaves the remaining $2n-4$ parts for agents 1 to $n-1$. Removing all the non-valuable parts for these agents, the remaining valuable parts of the cake are worth a_i to agent $i \in [n-1]$. Divide all valuations and entitlements by a_i for each $i \in [n-1]$. Then, this is equivalent to a cake with value 1 to every agent such that each agent's entitlement is $w'_i = 1/na_i$. If we select the a_i's carefully such that $\sum_{i\in[n-1]} w'_i = 1$ and the entitlements w'_i's are *generic*, then we can invoke Theorem 3.6 to show that the lower bound number of queries is in $\Omega(n \cdot 2^n)$.

Theorem 4.4. *Any algorithm that decides whether a connected strongly-proportional allocation exists for n agents with equal entitlements requires $\Omega(n \cdot 2^n)$ queries.*

The upper bound from Theorem 4.3 and the lower bound from Theorem 4.4 imply that the number of queries required to determine the existence of a connected strongly-proportional allocation is in $\Theta(n \cdot 2^n)$, even for agents with equal entitlements. The same tight bound also holds for *computing* such an allocation if it exists.

Theorem 4.5. *The number of queries required to decide the existence of a connected strongly-proportional allocation for n agents, or to compute such an allocation if it exists, is in $\Theta(n \cdot 2^n)$, even for agents with equal entitlements.*

5 Conclusion

We have studied necessary and sufficient conditions for the existence of a connected strongly-proportional allocation on the interval cake (Theorems 3.2 and 4.2). We have shown that computing this condition requires $\Theta(n \cdot 2^n)$ queries even for agents with equal entitlements (Theorem 4.5) or hungry agents with generic entitlements (Theorem 3.7), and $\Theta(n^2)$ for hungry agents with equal entitlements (Theorem 3.5). The same bounds hold for the computation of such an allocation if it exists.

A natural question that arose from our work is whether there is an algorithm that (asymptotically) attains the lower bound in (1) for hungry agents with entitlements that are neither generic nor equal.

Additionally, our work can be extended in the following ways:

- **Chores.** *Chore-cutting* is a variant of cake-cutting in which agents have *negative* valuations for every piece of the cake.
- **Beyond the unit interval.** We can consider cakes with more complex topologies, such as graphical cakes [4], tangled cakes [12], two-dimensional cakes [19], and pies [14].
- **Envy-freeness.** It is known that, in every cake-cutting instance, a connected envy-free allocation exists [21, 23]. What conditions are necessary and sufficient for the existence of a connected strongly-proportional allocation that is also envy-free?

Acknowledgements

This work was partially supported by the Hungarian Scientific Research Fund (OTKA grant K143858), by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under grant number 513023562, by NKFIH OTKA-129211, by the Israel Science Foundation grant 712/20, and by the Singapore Ministry of Education under grant number MOE-T2EP20221-0001. We thank Tassle and Neal Young for their helpful answers,[7] Warut Suksompong for his feedback on an earlier draft, and participants of the COMSOC online seminar and the anonymous ECAI 2024 reviewers for their comments. We are grateful to an anonymous ECAI 2024 reviewer for suggesting the solution to Case 2 in the proof of Lemma 3.1.

References

[1] Y. Aumann, Y. Dombb, and A. Hassidim. Computing socially-efficient cake divisions. *CoRR*, abs/1205.3982, 2012.
[2] J. Barbanel. Game-theoretic algorithms for fair and strongly fair cake division with entitlements. *Colloquium Mathematicae*, 69(1):59–73, 1996. URL http://eudml.org/doc/210327.
[3] J. B. Barbanel. Super envy-free cake division and independence of measures. *Journal of Mathematical Analysis and Applications*, 197(1):54–60, 1996.
[4] X. Bei and W. Suksompong. Dividing a graphical cake. In *Proceedings of the 35th AAAI Conference on Artificial Intelligence (AAAI)*, pages 5159–5166, 2021.
[5] G. Chèze. Envy-free cake cutting: A polynomial number of queries with high probability. *CoRR*, abs/2005.01982, 2020.
[6] L. Crew, B. Narayanan, and S. Spirkl. Disproportionate division. *Bulletin of the London Mathematical Society*, 52(5):885–890, 2020.
[7] Á. Cseh and T. Fleiner. The complexity of cake cutting with unequal shares. *ACM Transactions on Algorithms (TALG)*, 16(3):1–21, 2020.
[8] L. E. Dubins and E. H. Spanier. How to cut a cake fairly. *The American Mathematical Monthly*, 68(1P1):1–17, 1961.
[9] E. Elkind, E. Segal-Halevi, and W. Suksompong. Mind the gap: Cake cutting with separation. *Artificial Intelligence*, 313:103783, 2022.
[10] S. Even and A. Paz. A note on cake cutting. *Discrete Applied Mathematics*, 7(3):285–296, 1984.
[11] P. W. Goldberg, A. Hollender, and W. Suksompong. Contiguous cake cutting: Hardness results and approximation algorithms. *Journal of Artificial Intelligence Research*, 69:109–141, 2020.
[12] A. Igarashi and W. S. Zwicker. Fair division of graphs and of tangled cakes. *Mathematical Programming*, pages 1–45, 2023.
[13] Zs. Jankó and A. Joó. Cutting a cake for infinitely many guests. *The Electronic Journal of Combinatorics Volume 29, Issue 1*, 2022.
[14] Zs. Jankó, A. Joó, E. Segal-Halevi, and S. M. Yuen. On connected strongly-proportional cake-cutting. *CoRR*, abs/2312.15326, 2024.
[15] A. D. Procaccia. Cake cutting algorithms. In F. Brandt, V. Conitzer, U. Endriss, J. Lang, and A. D. Procaccia, editors, *Handbook of Computational Social Choice*, chapter 13, pages 311–329. Cambridge University Press, 2016.
[16] K. Rebman. How to get (at least) a fair share of the cake. *Mathematical Plums (Edited by R. Honsberger), The Mathematical Association of America*, pages 22–37, 1979.
[17] J. Robertson and W. Webb. *Cake-Cutting Algorithms: Be Fair if You Can*. Peters/CRC Press, 1998.
[18] E. Segal-Halevi. Cake-cutting with different entitlements: How many cuts are needed? *Journal of Mathematical Analysis and Applications*, 480(1):123382, 2019.
[19] E. Segal-Halevi, S. Nitzan, A. Hassidim, and Y. Aumann. Fair and square: Cake-cutting in two dimensions. *Journal of Mathematical Economics*, 70:1–28, 2017.
[20] H. Steinhaus. The problem of fair division. *Econometrica*, 16:101–104, 1948.
[21] W. Stromquist. How to cut a cake fairly. *American Mathematical Monthly*, 87(8):640–644, 1980.
[22] W. Stromquist. Envy-free cake divisions cannot be found by finite protocols. *the electronic journal of combinatorics*, 15(1):R11, 2008.
[23] F. E. Su. Rental harmony: Sperner's lemma in fair division. *American Mathematical Monthly*, 106(10):930–942, 1999.
[24] W. Suksompong. Constraints in fair division. *ACM SIGecom Exchanges*, 19(2):46–61, 2021.
[25] W. A. Webb. An algorithm for super envy-free cake division. *Journal of mathematical analysis and applications*, 239(1):175–179, 1999.
[26] G. J. Woeginger and J. Sgall. On the complexity of cake cutting. *Discrete Optimization*, 4(2):213–220, 2007.
[27] D. R. Woodall. A note on the cake-division problem. *J. Comb. Theory, Ser. A*, 42(2):300–301, 1986.

[7] https://cstheory.stackexchange.com/q/53901/9453