The Cutting Edge
An Introduction to Envy-Free Cake Cutting

Styopa Zharkov and Bharath Namboothiry

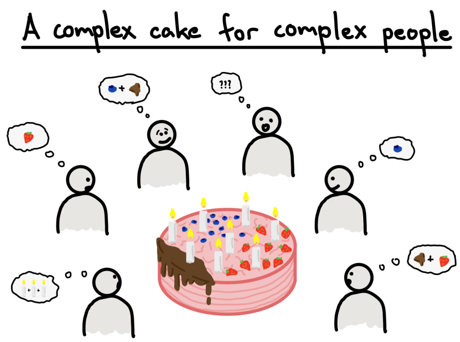
1 Introduction

Anyone with a sibling is acquainted with the divide-and-choose method for splitting food. One person divides the food into two parts that they think are equal and the other chooses which of the two they want. That way, no one will feel like they got the smaller half.

The problem becomes significantly more difficult when there are three people involved and even more difficult with n or more.

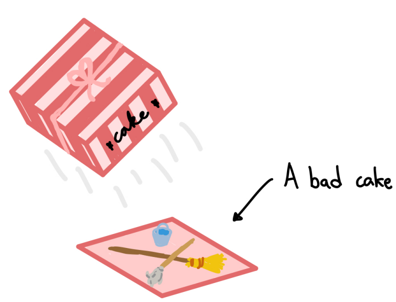
In this paper, we will look at how to split up food (here, a cake) between many people so that everyone is happy. There are several ways to define "everyone is happy". Here, we will say everyone is happy if they don't think anyone else's piece of cake is better than theirs (a property known as "envy-freeness").

The problem is more complex than simply dividing the cake into equal pieces, however. The cake could be fancy—it could have strawberries on one side, blueberries on the other and maybe chocolate somewhere else. Humans are also complicated. Some might like strawberries, some might not, and some might only love the pieces where exactly half is covered in chocolate. We will see how to formalize these ideas.



Is it possible to divide the cake fairly? If so, how do we do it? How much questioning and splitting would the people need to endure before getting their piece of cake? The answers to these questions are useful beyond settling fights between siblings. In fact, cake can be seen as a metaphor for any divisible resource. Ideas from the cake cutting problem can be applied to nations dividing land, servers allocating compute power, or employees scheduling shifts.

There are many variations of this problem. Maybe we want everyone to get one connected piece of cake and not many parts. Maybe the people don't really care for small differences, so we just need an approximately fair allocation. Maybe the cake is disgusting and no one wants to eat it, but it must be eaten out of respect to the host. That would be similar to dividing up chores between housemates. Perhaps the cake is actually a very crumbly cookie that is hard to divide evenly. Even with two people, that would be unfair to the person doing the splitting.

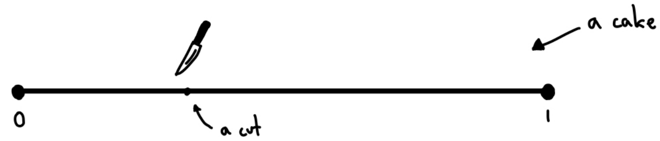


We will briefly explore some of these cases and see what questions we know how to answer.

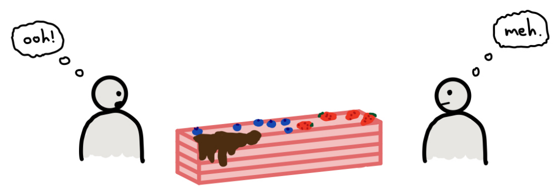
2 The Formal Model

Before we can prove anything, we need to formally define the ideas we are talking about. What does it mean for someone to like their piece of cake? What even is a cake?

To simplify our logic and calculations, we will assume that our cake is the line segment $[0, 1]$. To cut the cake is to simply mark a point on the line segment.



Definition 1: A person's *valuation* is a function from the space of possible pieces of cake to the real numbers. Note that the value of the same piece of cake can be different for different people. ◁



In this paper, we will make some mild assumptions about the valuations. We assume:

1. *Positive Definiteness.* The value of nothing or a single point of cake is 0 for everyone. Any segment of nonzero length has positive value for everyone (everyone likes cake).
2. *Additivity.* The sum of values of two pieces of cake is the value of the two pieces together[1].
3. *Continuity.* The valuations are continuous (crumbs aren't worth a lot).

There are some immediate consequences to these assumptions. The worthlessness of singular points means we do not have to worry about who gets the border when the cake is cut. Additivity gives us that having more cake can't make anyone worse off.

As mentioned in the introduction, the idea is that we want to allocate the cake to the people so that no one envies anyone else. This is what we'll consider to be a fair allocation.

Definition 2: We say an allocation of cake to people is *envy-free* if no one values someone else's cake more than their own. ◁

With these definitions, we can actually formalize what questions we're allowed to ask the people to decide how to split up the cake. This is important for us to be able to measure the "runtime" of our cake-cutting algorithms as the number of questions asked.

2.1 Robertson-Webb Query Computation Model

We will use is the Robertson-Webb query computation model. This model essentially allows for two kinds of questions (or "queries").

i) We can ask a person to say their value of a given interval. The interval need not be already cut off—the question can be hypothetical.

[1] This is the least mild of our assumptions. After all, ice cream with ketchup is definitely worse than ice cream or ketchup.

ii) We can ask a person to say how much of a piece of cake should be cut off for it to match a given value or say that it's impossible. More formally, we can give an endpoint a and a value v and ask for a second endpoint b such that the cake interval from a to b has value v. Note that as long as the value we ask for is no more than the value of the entire piece under consideration, such a cut is possible because of continuity of valuations. Again, the cut doesn't necessarily need to be made after the query is answered.

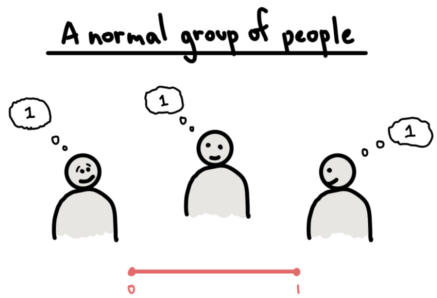
Let's look at what the divide and choose method looks like using this model.

Procedure 1 (Divide and Choose): Let A and B be two people. Consider the following algorithm to divide the cake.

1. Ask A, for the value of the entire cake. Call this value v_A.
2. Ask B for the value of the entire cake. Call this v_B.
3. Ask A to show a slice of value $v_A/2$ with one endpoint at 0. Call that slice I.
4. Ask B the value of I. If it's more than $v_B/2$, give I to B and the rest to A. Otherwise, give I to A and the rest to B. $\triangleleft$

This algorithm makes 4 queries. Person A thinks she got half of the cake and B thinks she got at least half of the cake, so neither envies the other. Thus, an allocation using the divide and choose procedure is envy-free[2].

Note that if all of the comparisons our algorithm makes are never between two different people[3], then we can normalize the value of the entire cake to be 1 for everyone. Not normalizing would add at most n extra queries because we can just ask everyone for their value of the entire cake and divide throughout. In the case of divide and choose, normalizing would let us get rid of the first two queries and replace v_A and v_B with 1. We'll often assume that the valuations are normalized and remember that we would need n extra queries to normalize them.



3 Disconnected Pieces

In some cases, we might want everyone to get one connected piece of cake. In others, we are fine with allocating two or more pieces to someone. So, we split the problem into two cases. In this section, we'll talk about the latter. Note that this case is only easier because every allocation that has connected pieces also satisfies the requirements for an allocation with disconnected pieces. We'll see both a lower and an upper bound for the number of queries needed to make an envy-free allocation with disconnected pieces.

It's not obvious that an envy-free allocation even exists for any number of people. However, we'll later show that an envy-free allocation exists even with connected pieces, which is a stronger statement, so let's delay that thought.

[2] We used additivity here. Otherwise, we wouldn't have known that the second half is just as valuable to A as the first half.

[3] This should be the case with any reasonable algorithm that tries to make an envy-free allocation because the definition of envy-freeness never compares values of different people. Intuitively, our algorithm shouldn't care if someone simply has stronger feelings about cake.

The number of queries required to find such an allocation is not known, however. For two people, the divide and choose protocol solves this problem. For three people, we'll present the Selfridge-Conway procedure. For four or more people, it was an open problem if a bounded procedure even exists until 2016, when Aziz and MacKenzie provided one[4]. As we'll see, the gap between the current known lower bound and upper bound is still very large, however.

First, let's look at Selfridge and Conway's procedure for three people.

3.1 Selfridge-Conway Procedure

The Selfridge-Conway procedure was the first envy-free discrete prodecure for three people, and served as a crucial starting point for tackling the n-person case. The strategy carries the same spirit as divide and choose, with a maximum of five cuts needed in its worst case. At a high level, there are two phases. In phase 1, the majority of the cake is cut into three pieces, one for each person. Through this process, a fourth piece will remain. In phase two, this fourth piece is split up into three more pieces and distributed among the people. The full procedure is as follows:

Procedure 2 (Selfridge-Conway): Begin with 3 people: A, B and C.

PHASE 1: Dividing most of the Cake

- A cuts the cake into, from their perspective, three equal pieces
- B identifies what they believes to be the best piece I, and trims that piece to make it have the same value as the second largest piece. Let's name the trimmed piece I_1 and the trim I_2.

  If B feels there are at least 2 best pieces, then C simply picks their favorite piece, then B picks one of their best pieces, then A takes the remaining.
- C chooses the best piece in their view (excluding I_2)
- B is forced to choose I_1 if it remains. If not, they choose the best remaining piece (excluding I_2)
- A is given the last remaining original piece, leaving the whole cake distributed except I_2

PHASE 2: Dividing the Trim

- I_1 is currently with B or C. Lets denote the person with I_1 by X_I and the other X_J
- X_J cuts the trim, I_2, into three equal pieces in their view
- X_I chooses a piece of I_2, which we can call L_1
- A chooses a piece of I_2, which we can call L_2
- X_J takes the final piece, which we can call L_3 $\triangleleft$

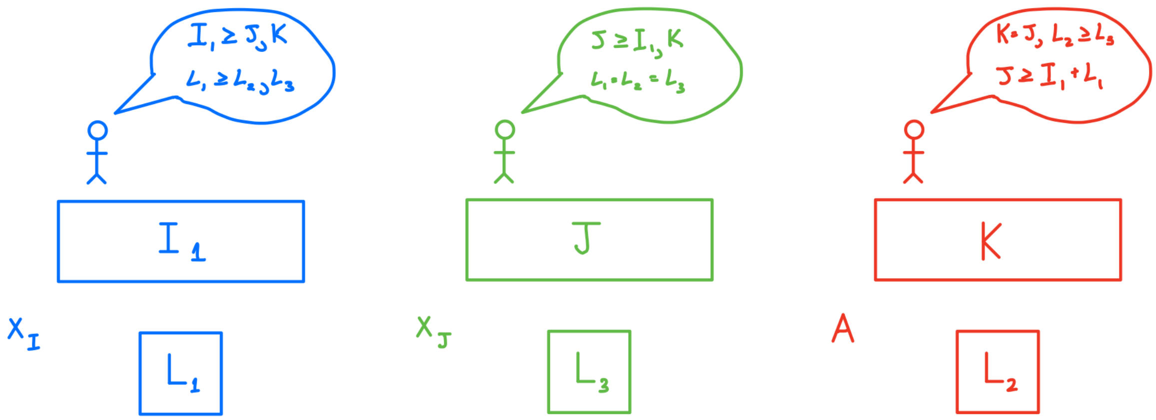
Analysis: This procedure is indeed envy-free. To review, X_I received I_1 and L_1, X_J received some original piece J and L_3, and A received some original piece K and L_2. So, let's go through the people one-by-one:

- X_I: If X_I is C, then they freely choose I_1 over J and K. If X_I is B, then I_1 in their eyes is tied for the best piece of cake. Either way, to X_I, we know $I_1 \geq J, K$. X_I also freely chose L_1 over L_2 and L_3, so X_I has no reason to envy any other people pieces.
- X_J: If X_J is C, then they freely chose J over I_1 and K. If X_J is B then J in their eyes must be the piece tied with I_1 as the best. Either way, to X_J, we know $J \geq I_1, K$. Also, since X_J is the one that cut up I_2, to them $L_3 = L_1 = L_2$. So X_J also has no reason to be envious.
- A: Since A make the first cut, they must view $J = K$ and $K = I_1 + I_2 \geq I_1$.

[4]Note that there did exist finite, but unbounded procedures before this. The Brams-Taylor protocol discovered in 1995 is the first of these.

- Since A picked L_2 before X_J picked L_3, she has no reason to envy X_J
- Recall that A views $K = I_1 + I_2$. Since L_1 is only part of the trimming I_2, X_I's pieces I_1 and L_1 have less value than K to A. So, she has no reason to envy X_I

The total positions, then, can be summarized in the following graphic:



Of course, if B observed a tie in A's initial cut, then C gets their pick, followed by B selecting one of the tied pieces, then A selecting one of their equally cut pieces (in which case, everyone is happy!). So, we have shown this cut is indeed envy free! As it turns out, the Selfridge-Conway procedure is also *proportional* (formal definition in section 5), meaning that each person receives $\frac{1}{3}$ of the entire cake's value. This is fairly straightforward to see, and can be left as an excercise for the reader.

3.2 Hardness Result

Now, let's look at a lower bound for the "complexity" of finding an envy-free allocation of a cake between n people according to the Robertson-Webb model.

Theorem 1: The number of queries it takes to find an envy-free allocation of a cake between n people (with disconnected pieces allowed) is $\Omega(n^2)$. $\triangleleft$

The proof is based on an analysis of what the algorithm could possibly know about the valuation of each person. Intuitively, if we have made a small amount of queries, then there's a small amount of points on the cake that the algorithm can work with, and we need enough points to have information on what interval to give to who. Let's formalize this.

Proof of Theorem 1: Let our cake be the interval $[0, 1]$ and let's assume that everyone's value of the entire cake is normalized to 1. Let $A_1, \ldots, A_n$ be the people participating.

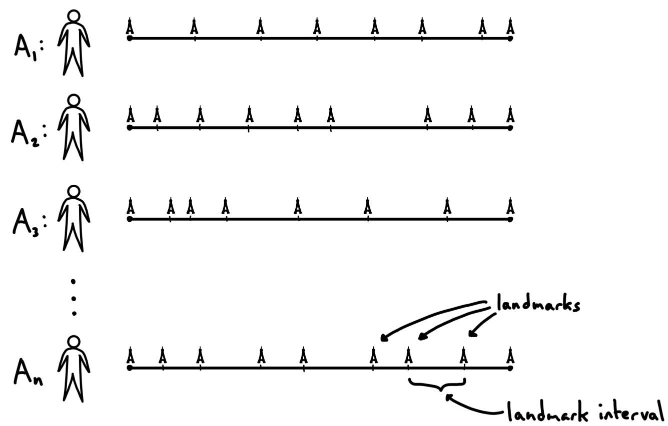
Each query either asks a person A_i for the value of an interval or asks to show an interval of a given value. In either case, at most two points are involved in the query (either as part of the question or as part of the answer). Let's keep track of the points we have information on for each person. Let L_i be the set containing 0, 1, and all points that were at some point involved in some query for A_i. We'll call the points in L_i the *landmarks* of A_i. These landmarks divide $[0, 1]$ into smaller intervals. Call these *landmark intervals.* Before any queries, when the landmarks are only 0 and 1, all we know is that A_i values the intervals $[0, 0]$ and $[1, 1]$ to be 0 and the interval $[0, 1]$ to be 1. We don't know anything about how the value is distributed on the interval. Similarly, when we have k landmarks that divide the interval into $k - 1$ landmark intervals, we cannot know how the value of each landmark interval is distributed within that interval because any distribution would give the same query results.

Once the cake is allocated, every person must value their piece at least as much as any of their landmark intervals. To see this, suppose for contradiction that A_i received a piece that she values less than some landmark interval I. Then, A_i didn't receive the entirety of I in her piece, so some other person A_j received part of I. As mentioned earlier, it could be that A_i's value of I is concentrated within the part that A_j

received, so A_i could envy A_j, which isn't possible with an envy-free allocation.

Now, suppose that the people's valuations of an interval are all simply the length of the interval. By the previous point, we know that no one can receive a piece that is shorter[5] than their longest landmark interval. It can't be that more than $n/2$ people got pieces that are longer than $2/n$, so at least $1-n/2 = n/2$ people got a piece of length no more than $2/n$. For each of these people, all of the landmark intervals must have length at most $2/n$, so there must be at least $n/2$ landmark intervals, so there must be at least $n/2$ landmarks.

Remember that each query can only add at most 2 new landmarks for to the person queried. Since we have at least $n/2$ people, each with at least $n/2$ landmarks, the total number of queries required to make an envy-free allocation must be $(n/2 \cdot n/2)/2 = n^2/8 = \Omega(n^2)$. □



3.3 Aziz and MacKenzie's Bounded Procedure

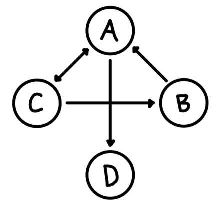
We've talked about the three-person case, and we saw a lower bound for the number of queries needed for the general n-people case. As mentioned earlier, the existence of a bounded procedure for four or more people was only recently proven by Aziz and MacKenzie. Unfortunately, Aziz and MacKenzie's description of the procedure and the proof are too complex to fit in this paper, so we'll only give an intuition for how it works. First, they gave a procedure for four people, and weeks later, they extended the algorithm to any number of people.

Procedure 3 (Aziz and MacKenzie Idea): Aziz and MacKenzie present first an algorithm for a partial envy free allocation for 4 people, in which each person *dominates* two other people (see algorithm 3 of [1]). In this context, the word dominates implies that a person will never envy another, no matter how much more cake they get. (For example, if X dominated Y in a partial allocation with some remaining unallocated cake β, then even if Y gets the entirety of β, X will still not envy Y.) If there is no remaining portion β, the whole cake has been allocated and an evny-free solution has been reached. If some cake remains, they construct an envy-free complete allocation of the remaining piece as follows:

Analysis: Now, let's argue why the above works. Recall that A dominates C and D. If B also dominates C and D, we know we can give all of β to C and D without causing envy in the other people. Now, since the task is to split up a piece of cake among two people an not cause envy, we simply use divide and choose, and we are done!

Assuming B only dominates A and one other person (let's say D without loss of generality), there are two cases. First, if C also dominates D, then we can freely give all of β to D, as they are dominated by everyone, and we are done. Second, if C does not dominate D, they must dominate A and B. In this case, the domination graph looks as follows:

[5] Pieces, can be disconnected, but we can still define the length of the piece to be the sum of lengths of all of the connected parts in that piece.



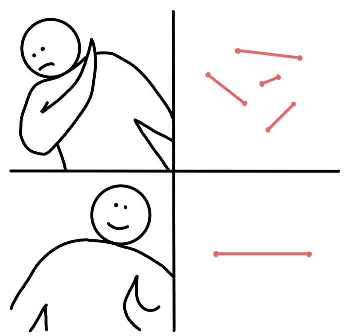
Here, D can cut up β into four equal pieces (from their perspective). Then, the people pick their favorite pieces in the order A, B, C. In this ordering, no person is preceded by someone they do not dominate, so envy-freeness is preserved. Finally, the remaining piece is given to D, but since D was the one who divided β equally, they have no reason to envy any other person. So, we are done!

In all cases then, we arrive at an envy-free solution!

Just weeks later, the authors converted their procedure to the n-person case [2]. In spirit, this algorithm transforms partial envy-free allocations into complete ones recursively, arriving at a total allocation in bounded time. Of course, the algorithm is far from efficient. The maximum number of queries that their algorithm needs is bounded by $n^{n^{n^{n^{n^{n}}}}}$. Finding a faster bounded solution (let alone near the $\Omega(n^2)$ lower bound), remains a very exciting open problem.

4 Connected Pieces

Now, let's switch over to the case where we want everyone to have one connected piece of cake. This applies to situations like dividing a piece of land between countries where the territory should be connected or diving computer memory where each allocated piece of memory should be contiguous[6].



First, we'll prove that an envy-free allocation exits, which isn't obvious. Next, we'll show that in contrast to the disconnected pieces problem, there is no bounded procedure to find an envy-free allocation using the Robertson-Webb model. We'll then look at an alternative model that does let us create a bounded procedure for dividing the cake.

4.1 Proof of Existence

For this proof, we will need to be rigorous about what it actually means to "cut a cake". Specifically, we will insist that a cake can be cut by $n-1$ planes, each parallel to a given planes. These possible cuts then can be represented as numbers in the interval $[0, 1]$, and possible divisions can be represented by vectors $x = (x_1, \dots x_{n-1})$ such that $0 \le x_1 \le \dots \le x_{n-1} \le 1$. For convention, we can think of $x_0 = 0$ and $x_n = 1$, so we have planes that bind every slice. All possible divisions form a compact set in $\mathbb{R}^{n-1}$, which we can refer to as the *division simplex*

$$S = \{(x_1, \dots x_{n-1}) | 0 \le x_1 \le \dots \le x_{n-1} \le 1\}$$

And let S_i be the set of divisions where the ith piece is empty. Recall also that we assume each person's cake valuation is continuous. More formally, we assume that the the choice for the jth person is based on a

[6]Plus, surely no one enjoys a pile of crumbs of cake as much as a whole piece.

real-valued continuous valuation function f_j, which give the value of the ith piece in terms of $x = (x_1, \dots x_n)$. So the value of the ith piece to the jth player is denoted by $f_j(x, i)$.

For a given x, or course, we say j prefers the ith piece if they do not envy any other pieces (if $f_j(x, i) \geq f_j(x, k)$ for all k). We meet the criteria of envy-freeness if every person can be given a preferred piece. Finally, recall that every no person cal prefer an empty piece of cake. With these clarifications, we can state the following theorem:

Theorem 2: For any cake and any group of n people with positive definite, continuous, and additive valuations, there exists an envy-free allocation of the cake with connected pieces.

Proof of Theorem 2: For each i, j let A_{ij} be the set of divisions $x \in S$ for which the j-th player prefers the i-th piece. From the continuity of each f_j, we know that each A_{ij} is closed. For each j, the sets A_{ij} cover S. Since we assume that no person prefers an empty piece of cake, we infer that for each i, j, $A_{ij} \cap S_i = \varnothing$.

Now, we can let $B_{ij} = \cup_{k \neq i}(S - A_{kj})$, which can be understood as the set of divisions where the jth player prefers *only* the ith piece. Next, define $U_i = \cup_j B_{ij}$, which can be understood as the set of divisions for which *some* player prefers only the ith piece. Each B_{ij} is open in S, so we infer that each U_i is open in S. From here, we examine two cases:

- The Usual Case: The sets U_i cover S

  In this case, we rely on the following lemma:

  Lemma 1: Suppose a $(n-1)$-simplex S is covered by n open $U_1, \dots U_n$, such that no U_i intersect with the corresponding face S_i. Then, the common intersection of $U_1, \dots, U_n$ is nonempty.

  The proof of this lemma can be found in [3]. Since we know there to be at least one $x \in \cap_i U_i$, we know there is some division x such that every every piece will be the unique preferred piece for some person. So, we know there must be an envy-free assignment.

- The Unusual Case: The sets U_i do not cover S

  We consider this case to be unusual because it depends on a coincidence. If $x \in S$ is not in any U_i, it must leave *every* person indifferent to two or more "best" pieces. Of course, this is very possible (for example, when every player has the same exact valuation function). The strategy here will be to approximate the sets A_{ij} with the sets A'_{ij} for which this coincidence doesn't occur. Of course, this would put us back in the usual case, and we know that there is an envy-free allocation if the player's preferences were described by A'_{ij}. As the approximations improve, the solutions converge to an envy-free allocation of the original preferences.

  The argument above relies wielding a set of n irrational numbers that are linearly independent over rationals to divide up S into sets that begin to look like A_{ij} over time. The technical details of this are somewhat out of the scope of the paper, but can be found in [3] $\square$

4.2 Hardness Result

Although an envy-free allocation with connected pieces always exists, it's hard to find even for three people. It turns out that with the two query options we have, the problem is in a sense equivalent to finding a real number on the interval $[0, 1]$ using yes or no questions, which is not possible in finite time. We'll prove this with an assumption of one small lemma that is surprisingly tedious to prove.

Theorem 2: An envy-free allocation between 3 people cannot be found in finite time with the Robertson-Webb query model. $\triangleleft$

The idea for the proof is to construct the valuations in such a way that the cuts must be at very specific points. Then, no matter what the algorithm is, as it makes queries, we'll be able to slightly modify the valuations so that none of the past answers would change, but the points at which the cuts need to be made do[7].

[7] If you've ever played the game "Contact", this is similar to the strategy of changing your word to something that starts with the same letters if the others are about to guess it.

First, let's see what valuations we'll use to make the cuts required to be at a specific point.

Definition 3: A *rigid measure system* (RMS) is a system of 3 valuations on a $[0, 1]$ cake that is constructed as follows. Let s, t, x and y be real numbers satisfying the following points:

- $0 < s < t < 1$
- $0 < x < y < 1$
- $s + 2t = 1$

Construct three valuations v_A, v_B, and v_C for people A, B, and C such that

1. For any interval $[a, b]$ where $a \neq b$, we have that $v_X((a, b))/(b - a)$ is strictly between $\sqrt{2}/2$ and $\sqrt{2}$ for any person X. In other words, the density of the valuations is always between $\sqrt{2}/2$ and $\sqrt{2}$.
2. The valuations of the intervals $[0, x]$, $[x, y]$, and $[y, 1]$ follow the following table:

	[0, x]	[x, y]	[y, 1]
A	t	t	s
B	s	t	t
C	t	s	t

◁

Now, let's prove a couple of lemmas about rigid measure systems that we will use.

Lemma 1: Let I and J be two pieces of cake. Let X and Y be any two people with valuations from an RMS. Then, $v_X(I) \geq 2v_X(J) \implies 2v_Y(I) > v_Y(J)$. ◁

Intuitively, this just means that the valuations of cake pieces can't differ too much between different people. The proof is a simple consequence of the bounds on the density of valuations in the definition of an RMS.

Proof of Lemma 1: Let us denote the lengths of I and J by $|I|$ and $|J|$. Suppose $v_X(I) \geq 2v_X(J)$ Then,

$$\begin{aligned} 2v_Y(I) &> 2 \cdot \sqrt{2}/2 \cdot |I| \\ &= \sqrt{2} \cdot |I| \\ &> v_X(I) \\ &\geq 2v_X(J) \\ &> 2 \cdot \sqrt{2}/2 \cdot |J| \\ &= \sqrt{2} \cdot |J| \\ &> v_Y(J), \end{aligned}$$

which is what we wanted to prove. □

Now, let's prove that the cuts must be at specific points, like we wanted. For the proof, we'll just look at the possible cases for where the cuts can be.

Lemma 2: Any envy free allocation with connected pieces under an RMS must have cuts at x and y. ◁

Proof of Lemma 2: Since the allocation must have three connected pieces, there must be two cuts made. Let the leftmost cut be at $\bar{x}$ and the rightmost cut be at $\bar{y}$. If the cuts are not made exactly at x and y, there are four possible cases, all of which can't be envy-free.

1. If $\bar{x} \geq x$ and $\bar{y} \leq y$, and one of the inequalities is strict, then everyone values either the right piece or the left piece more than the middle, so whoever gets the middle will be envious.

2. If $\bar{x} \leq x$ and $\bar{y} \geq y$, and one of the inequalities is strict, then both A and B prefer the middle piece over any other piece, so whoever won't get it out of the two will be envious of the other.

3. If $\bar{x} > x$ and $\bar{y} > y$, then both A and C prefer the left piece to the right piece, so neither of them should get the right piece, so B should get the right piece. Since B shouldn't envy for the middle piece, this would mean that B's value of $[x, \bar{x}]$ is at least twice as much as her value of $[y, \bar{y}]$. By Lemma 1, this means that both A and C value $[y, \bar{y}]$ less than twice their value of $[x, \bar{x}]$. So, both A and C must like the left piece more than they like the middle piece, so whoever gets the middle piece will be envious.

4. If $\bar{x} < x$ and $\bar{y} < y$, then the logic is analogous to the previous case.

As we can see, the only way to make an envy-free allocation is to cut at x and y. $\square$

The final lemma that we need is the one we won't provide a proof for.

Lemma 3: Consider any RMS. Let X be a participating person, and let $\varepsilon > 0$. There exists a different RMS with different cutpoints $x' \neq x$ and $y' \neq y$ such that v_X is exactly the same and the valuations of the other people are exactly the same except for on an ε-ball around x and y. $\triangleleft$

In other words, we can slightly shift the x and y without changing one of the people. If ε is small enough, then we also won't change any of our previous answers. With this thought, let's prove the theorem.

Proof of Theorem 2: Suppose we have a procedure that runs in finite time. Let's construct a system of valuations that fools the procedure into making the wrong cuts. The construction works as follows.

1. Start with any RMS. Say, with $s = 1/4, t = 3/8, x = 1/3$, and $y = 2/3$ and uniform density of valuations on the three parts.

2. If the procedure asks about points other than x and y, let it run.

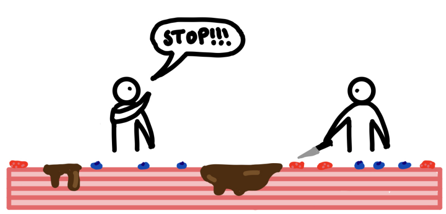
3. Any time the procedure asks person X about either x or y, apply Lemma 3 to switch our RMS for one with a different x and y, but so that none of the previous queries would have been answered differently. Note that we can always make ε small enough for this since the number of points asked about so far must be finite.

4. Once the procedure finishes, if it tries to make a cut at the latest x and y, use Lemma 3 again to ensure that the cut isn't made at the x and y.

The resulting RMS from this construction will fool the procedure into making a cut not at the x and y. By Lemma 2, this means that the allocation isn't envy-free. So, no finite procedure can exist. $\square$

4.3 Let's Move Knives Instead

We saw in the previous section that no finite algorithm for connected pieces exists, but let's cheat a little and introduce the moving knives model. The idea is that now we'll allow for queries of the type "I move the knife over the cake and you tell me when to stop" as well as "Move the knife over the cake until I tell you to stop". Since our cake is a line segment, we assume that the knives are oriented perpendicular to the cake as they move. Procedures that follow this model aren't truly finite because each person is making an infinite and even uncountable number of valuations as a knife moves over an interval of the cake.



Most moving knife procedures take the following form. A referee that slowly moves a knife from 0 to 1. Every person is given their own knife and also moves it over the cake simultaneously. At some point someone yells "cut", and a certain subset of the knives make the cut where they are. This could be repeated a finite number of times before the pieces are allocated.

4.4 Stromquist Moving-Knives Procedure

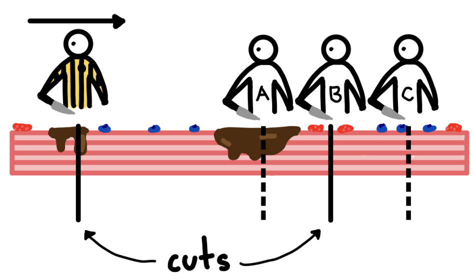
As an example, let's look at the Stromquist Moving-Knives Procedure. This procedure lets us find an envy-free allocation with connected pieces for three people, something that we showed wasn't possible with the Robertson-Webb model.

Procedure 4 (Stromquist): Let A, B, and C be the participating people. We give every participant a knife. A referee slowly moves a knife over the cake starting at 0 and going to 1. We ask the participants to continuously move their knives on the right side of the referee knife. At any point in time, a participant is allowed to yell "cut". Once someone does,

1. The referee and the middle of the three participant knives make a cut.
2. The left piece is given to the person who yelled.
3. The middle piece is given to the participant with the leftmost knife who doesn't have a piece yet.
4. The right piece is given to the remaining participant.

If two people yell to cut at the same time, pick one of the two to be the first arbitrarily.

The strategy for each participant is to hold their knife so that they divide the cake to the right of the referee exactly in half according to their valuations. As soon as the referee knife reaches a point where the piece to the referee is equal to the piece they would receive if someone else shouted to cut, they should shout to cut.



In other words, the left participant should yell cut when the middle piece has the same value as the left piece[8]. The middle participant should yell cut when all three pieces have the same value. The right participant should yell cut when the right piece has the same value as the left piece. ◁

Now, let's look at why this procedure actually makes the allocation envy-free as long as the people follow the strategy.

Analysis: Let's carefully analyze each person.

- Whoever yelled to cut can't envy either of the two pieces because they yelled to cut when the left piece is equal to the piece they would have gotten if someone else yelled to cut and at least as big as the third piece.
- Neither of the people who didn't yell to cut could envy the person that did because otherwise they would have yelled to cut earlier.
- The people who didn't yell cut can't envy each other because either they made the cut or their piece contained their knife, so they value their piece at least as much as the others.

Thus, our allocation is envy-free. □

5 Variants

We saw some of what is known about making envy-free allocations of line segment shaped cakes using the Robertson-Webb query model, but life is more complicated than that. This paper wouldn't be complete without a discussion of the different variants of the cake-cutting problem. In this section, we'll briefly explore some of these variants without going deep into any proofs.

[8] When we say piece here, we are supposing that the middle participant were to make a cut now.

5.1 Approximations

Envy-free cake allocation has proven to be an exceedingly difficult problem. However, is some contexts, we might be fine with an *approximate* solution.

In the context of cake cutting, there are two ways to define approximation - in length or value. Length based approximations are natural in many settings. For example, if our cake is a stretch of land, perhaps the recipients of the land agree that a difference of 0.1 meters does not matter to them. So, we are free to search for a partition that is within a 0.1 neighborhood of an envy-free partition. Similarly, in the case of value, imagine that the recipients only feel envy when their land is worth $>$ \$100 less than someone else's land. Then, we are free to search for a partition that is roughly envy-free, with small discrepancies that don't matter to the parties. In both cases, the problem feels more approachable, yet still has computational real world applications.

5.2 Free Disposal

An implicit constraint we have placed on ourselves in both the disconnected and connected settings is that we must actually allocate the entire cake, and not waste any. Consider if we relaxed this, and allowed ourselves to freely dispose of some portion of the cake (while of course, aiming to minimize the amount of cake we waste). Would the problem get any easier?

Our gut feeling is yes. Both the Selfridge-Conway procedure and the Aziz-MacKenzie Procedure are able to quickly generate envy-free allocations for *most* of the cake, and then spend a lot of time cleverly dividing up the remaining piece. If we could ignore the latter step, perhaps simpler solutions are within reach.

5.3 Other Measures of Goodness

Envy-freeness is only one way do define a "good" allocation. We can also look at some other definitions.

Definition 4: An allocation of cake between n people is *proportional* if everyone's value of the piece they got is at least a $1/n$ fraction of their value of the entire cake. $\triangleleft$

In other words, everyone thinks they got their fair share of cake. With additive valuations, every envy-free allocation is proportional[9]. So, proportional allocations always exist.

We saw that it's not possible to find an envy-free allocation with connected pieces in finite time. However, the *last diminisher procedure* neatly finds a proportional allocation with connected pieces in $O(n^2)$ queries according to the Robertson-Webb model. The idea for the algorithm is as follows.

Procedure 5 (Last Diminisher): Let's say there are n people names $A_1, A_2, \ldots, A_n$. First, we ask A_1 to show us how far we need to cut from 0 to get a $1/n$ fraction of the cake. Then, we ask A_2 to decrease how far we have to cut if she thinks it's more than a $1/n$ fraction. Then, we ask A_3 to decrease the (potentially) new value if she thinks it's still too far. We continue doing this until all n people have been asked. Then, we cut the cake at the last value and give the piece to whoever was the last person to change the value. Now, we recurse on the remaining $n-1$ people with the remaining cake until there is only only one person left, to whom we give the last piece. $\triangleleft$

Intuitively, this algorithm works because any time someone gets a piece, they made the last change, so they view their piece as at least a $1/n$ fraction. The last person must also get at least a $1/n$ fraction in their eyes because they must have viewed every other piece of being no more than a $1/n$ fraction. Note that this doesn't dive us an envy-free allocation because the first person to get cake might envy one of the later people if the valuations disagree.

We might want our allocation to be "the best" possible in some sense. For that, we give the following alternative measure of goodness.

Definition 5: An allocation is *Pareto-optimal* if there is no other allocation that makes someone better off and no one worse off. $\triangleleft$

[9] We can see this by looking at the contrapositive. If your piece is less than a $1/n$ fraction of the whole thing, then the other $n-1$ people are dividing more than a $(n-1)/n$ fraction between themselves, so someone must have gotten more than a $1/n$ fraction, so you envy them.

Pareto-optimality by itself isn't a very interesting cake-cutting goal. Giving the entire cake to someone is a Pareto-optimal allocation because any other allocation would take away from that person. However, requiring both Pareto-optimality and another fairness measure like evy-freeness can give some interesting results.

An intuitive way to try to show that an envy-free Pareto-optimal allocation exists is to take the envy-free allocation and if it's not Pareto-optimal, take the allocation that's "better" than it. Unfortunately this doesn't work because the better allocation might no longer be envy-free. Such an allocation does exist, however. The proof follows from a result called Weller's Theorem directly [4].

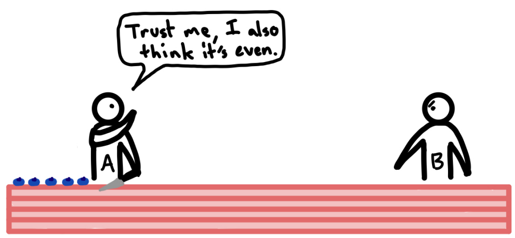
5.4 Truthfulness

In the previous section, we talked about restrictions on the resulting allocation. Here we will mention a fairness restriction on the algorithm and not the allocation.

Definition 6: A cake-cutting procedure is *truthful* if no person can lie about their values to obtain a better piece of cake. ◁

Essentially, a procedure is truthful if you can't "game the system". The motivation for wanting our procedures to be truthful is clear.

The divide and choose procedure isn't truthful. Suppose A and B are dividing a cake that is mostly plain, but there's a few blueberries in one spot. Let's say A likes all cake equally and doesn't care for blueberries and B absolutely loves blueberries and doesn't like the plain cake as much. Then, if A is the divider, she can lie about her valuation and claim that an even split would have a very small piece with all of the blueberries. Then, B would pick the small piece and A would get the large remaining piece, which is better for her than if she didn't lie and just split the cake evenly.



As we can see, there are numerous possible definitions for fairness and even more combinations of them. In most cases, little is known about the bounds on the runtime of algorithms to find them.

5.5 Other Cake Shapes

Of course, not all cakes are line segments. We can also consider other shapes. We can consider the case when our cake is a square and we're only allowed to make rectangular slices. Or, we could ask that everyone gets a square piece but allow for free disposal (because it's not possible to divide a square into two squares, for example). Similarly, we can extend this to any dimension[10]. Reasoning about general cake shapes with any type of cuts turns out to be incredibly difficult.

A newer, but well-studied variation is if our cake is a circle and we're only allowed to make radial cuts. This is called a pie. Pie-cutting is equivalent to dividing an interval where the endpoints are connected. So, any interval cake cutting algorithm is also a pie-cutting algorithm because any initial cut turns a pie into an interval. The interesting question lies in the efficiency of pie-cutting algorithms.

5.6 Disgusting Cakes

Another well-studied variant is when no one wants cake (i.e. the valuations of a nonzero interval of cake are negative. This could be applied in the problem of dividing chores or the problem of splitting the work of a project between the contributing organizations. We can also define envy-freeness in the same way and ask

[10] Although 20-dimensional cake-cutting sounds cool, there currently aren't many applications.

the same questions as with normal cakes.

There is an analog to the Selfridge-Conway protocol, but for bad cakes. It was first presented by Reza Oskui and requires a few extra cuts.

The first discrete procedure for envy-free bad cake division for any number of people was presented in 2009 by Peterson and Su. This procedure is finite, but not bounded (i.e. the procedure ends, but there's no maximum number of queries for a given n).

As for connected pieces, we know that an envy-free allocation of a bad cake with connected pieces always exists by a proof similar to that of normal cakes. There are several moving knife procedures to find this allocation.

5.7 Hard to Cut Cakes

Another interesting variant is to consider the very practical problem of inaccuracy. When dividing a cookie between two people using the divide and choose procedure, it's common to want to be the second person because a cookie is hard to split exactly in half—it's usually an uneven split.

Although we can't reason about exact envy-freeness as easily, we can still formalize this idea by assigning a random distribution to where the cut is going to be based on where we try to make it. Then, we can try to make the allocation as fair as possible in some sense.

This isn't a trivial question even if everyone's valuations are uniform and there are only two participants. One way to mitigate the unfairness towards the divider is to first have a participant divide the cookie into two, then have the other pick a half. Then, each participant can divide their half into two and pick a quarter from the other. This results in a more fair allocation, although the pieces aren't connected.

If we start considering truthfulness, then we have an interesting duality here. On one hand, you want to be first to take advantage of the non-truthfulness. On the other, you want to be second to have randomness on your side. Which is better depends on how much you know about your opponent and how wide the distribution for a cut is.

6 Conclusion

As a review, we introduced a formal model in which we can study the envy-free distribution of a cake, and a computational model within which we can develop algorithms. We defined first the *disconnected case*, where we are fine giving people multiple disconnected pieces of cake. With this freedom, we discussed algorithms for 3, 4, and n people, and presented a lower bound query complexity of discovering an envy-free allocation.

In the harder connected case, we proved the existence of an envy-free allocations of cake, together with a hardness result that forbids finite-query algorithmic solutions. In face of this result, we introduced a new type of "moving knives" query, which works around this result and allows for "finite" procedures.

Envy-free cake cutting research is far from complete. There remains a huge gap between the known upper and lower bounds for disconnected pieces. Each of the variants we discussed has unique practical use-cases, and presents a new line of yet-to-be-seen research. We really enjoyed learning about these problems, and are excited to see where computer scientists and mathematicians take their cakes from here!

References

[1] Haris Aziz and Simon Mackenzie. A discrete and bounded envy-free cake cutting protocol for four agents, 2016.

[2] Haris Aziz and Simon Mackenzie. A discrete and bounded envy-free cake cutting protocol for any number of agents, 2017.

[3] Walter Stromquist. How to cut a cake fairly. *The American Mathematical Monthly*, 87(8):640–644, 1980.

[4] Dietrich Weller. Fair division of a measurable space. *Journal of Mathematical Economics*, 14(1):5–17, 1985.