The Query Complexity of Cake Cutting

Simina Brânzei∗ Noam Nisan†

Abstract

We study the query complexity of cake cutting and give lower bounds for computing approximately envy-free, perfect, and equitable allocations with the minimum number of cuts. These lower bounds are tight for three players in the case of envy-free allocations and for two players in the case of perfect and equitable allocations. We also formalize moving knife procedures and show that a large subclass of this family, which captures all the known moving knife procedures, can be simulated efficiently with arbitrarily small error in the Robertson-Webb query model. Our query complexity upper bounds for envy-free allocations among three players and perfect allocations between two players rely on simulating approximately two moving knife procedures due to Barbanel-Brams and Austin, respectively.

1 Introduction

We study the classical cake cutting problem due to Steinhaus [31], which captures the division of a heterogeneous resource—such as land, time, mineral deposits, fossil fuels, and clean water [25]—among several parties with equal rights but different interests over the resource. This model has inspired a rich body of literature in mathematics, political science, economics, and computer science [27, 6, 22, 9], from which several fundamental notions of fairness emerged together with protocols designed for achieving them.

Mathematically, the problem is to divide the cake, which is the unit interval, among a set of $n$ players with valuation functions that are induced by probability measures over $[0, 1]$. The goal is often to compute an allocation of the cake such that each player receives a piece they are content with. A major challenge is that the preferences of the players are private, and fair allocations can only be found when enough information has been revealed. An example of a cake cutting protocol is Cut-and-Choose, which dates back to more than 2500 years ago when it appeared in written records in the context of land division. Cut-and-Choose can be used to obtain an envy-free allocation among two players, e.g. Alice and Bob, that are trying to cut a birthday cake with different toppings. Alice is instructed to cut the cake in two pieces that are of equal value to her, after which Bob picks his favorite piece among the two, while Alice takes the remainder.

∗Hebrew University of Jerusalem. E-mail: simina.branzei@gmail.com.
†Hebrew University of Jerusalem. E-mail: noam@cs.huji.ac.il.
More generally, classical cake cutting protocols operate in a query model, in which a center that does not know the players asks them questions about the values of different pieces until it manages to extract enough information for computing a fair division. Two of the prominent fairness notions, envy-freeness and proportionality, have been the subject of in depth study from a computational point of view. Proportionality requires that each player gets their fair share of the resources, which is their total value for the whole cake divided by the number of players, while envy-freeness is a notion based on social comparison and stipulates that no player should want to swap their piece with that of any other player. The two notions are not necessarily comparable, since envy-freeness can be trivially achieved by throwing away the entire resource, while this is not true for proportionality. However, when the entire cake must be allocated, envy-freeness implies proportionality and is surprisingly hard to achieve. The problem of finding an envy-free cake cutting protocol was suggested by Gamow and Stern in 1958 [20], and solved by Selfridge and Conway for three players (cca. 1960, see, e.g., [27, 6]) and by Brams and Taylor 35 years later [8] for any number of players. Unfortunately, from a computational point of view, the Brams and Taylor protocol has the property that its runtime can be made arbitrarily long by setting up the valuations of the players appropriately. In 2016, Aziz and Mackenzie [3] announced the first bounded envy-free cake cutting protocol for any number of players, where bounded means that the number of queries is only a function of the number of players, and not of the valuations.

The query complexity of proportionality is well understood. The problem of computing a proportional allocation with connected pieces can be solved with $O(n \log n)$ queries in the Robertson-Webb model using the Even-Paz protocol [19], with a matching lower bound given by Woeginger and Sgall for connected pieces [37] that was extended to any number of pieces by Edmunds and Pruhs [18]. In contrast, for the query complexity of envy-free cake cutting a lower bound of $\Omega(n^2)$ was given by Procaccia [24], and an upper bound of $O\left(\frac{n^2}{\log n}\right)$ by Aziz and Mackenzie [3]. Envy-free allocations with connected pieces have been known to exist for very general valuation functions, which are not necessarily induced by probability measures (see, e.g., Simmons [30], Stromquist [32] for a proof based on a topological lemma about intersection of sets, and Su [34] for a proof based on Sperner’s lemma). However, envy-free allocations with connected pieces cannot be computed in the Robertson-Webb model (Stromquist [33]) except for specific classes of valuations (such as polynomial functions [10]). Upper and lower bounds on the query complexity of approximate envy-free cake cutting with connected pieces, but for a general utility model where the valuations are arbitrary functions not necessarily induced by probability measures, were given by Deng, Qi, and Saberi [15]. Neither the upper bounds nor the lower bounds in [15] are tight in the standard cake cutting model.

A third notion of fairness is known as equitability, in which each player must receive a piece worth the same value. Equitable and proportional allocations with connected pieces were shown to exist by Cechlarova, Dobos, and Pillarova [12]. The computational complexity of approximate equitability was investigated by Cechlarova and Pillarova, who gave an upper bound of $O\left(n \left(\log n + \log \epsilon^{-1}\right)\right)$ for computing an $\epsilon$-equitable and proportional allocation with connected pieces for any number of players, while Procaccia and Wang [26] showed a lower bound of $\Omega\left(\frac{\log \epsilon^{-1}}{\log \log \epsilon^{-1}}\right)$ for finding an $\epsilon$-equitable allocation (not...
necessarily connected) for any number of players.

More stringent fairness requirements are also possible, such as the necklace splitting problem, for which the existence of fair solutions was established by Neyman [23], with a bound on the number of cuts given by Alon [1], the more general notion of exact division \footnote{The problem of exact division is the following: given a cake with \( n \) players and target non-negative weights \( w_1 \ldots w_k \), find a partition \( A = (A_1, \ldots, A_k) \) such that \( V_i(A_j) = w_j \) for each player \( i \) and every piece \( A_j \).}, which generalizes necklace splitting and was shown to exist by Dubins and Spanier [16], and the competitive equilibrium from equal incomes, the existence of which was determined in the cake cutting model by Weller [36].

### 1.1 Our Contribution

In this paper we study the query complexity of cake cutting in the standard (Robertson-Webb) query model [37] for several fairness notions, namely envy-free, perfect, and equitable allocations with the minimum number of cuts. Such allocations are known to exist on any instance, but several impossibility results preclude their computation in the Robertson-Webb query model [33, 13]. Nevertheless, the computation of approximately fair solutions is possible and we state a general simulation result implying that for each \( \epsilon > 0 \), a number of queries that is polynomial in \( n \) and \( 1/\epsilon \) suffices to find \( \epsilon \)-fair allocations for several fairness concepts, such as \( O(n/\epsilon) \) for connected envy-free and \( O(n(k-1)/\epsilon) \) for \((\epsilon, k)\)-measure splittings.

Moreover, we give several lower bounds for computing approximately fair allocations, showing that finding an \( \epsilon \)-envy-free allocation with connected pieces among three players, a connected \( \epsilon \)-equitable allocation for two players, and an \( \epsilon \)-perfect allocation with two cuts for two players requires \( \Omega \left( \log \frac{1}{\epsilon} \right) \) queries. The lower bound for envy-free allocations extends to any number of players. We also give upper bounds for these notions, which in the case of envy-freeness for three players and perfect allocations for two players are \( O(\log \frac{1}{\epsilon}) \) and rely on approximately simulating in the Robertson-Webb model two moving knife procedures, due to Barbanel and Brams [5] and Austin [2], respectively. An upper bound of \( O \left( \log \frac{1}{\epsilon} \right) \) for computing a connected \( \epsilon \)-equitable allocation for two players was given by Cechlarova and Pillarova [13].

Finally, we formalize the class of moving knife procedures, which was previously viewed as disjoint from the Robertson-Webb query model, and show that any moving knife procedure with a fixed number of players and knives can be simulated in \( O(\log \frac{1}{\epsilon}) \) queries in the Robertson-Webb model when the players have valuations that are bounded (from above and below) and the knife operations satisfy Lipschitz continuity. This simulation immediately implies that all the known moving knife procedures, such as the procedures designed by Austin, Barbanel-Brams, Stromquist [32], Webb [35], and Brams-Taylor-Zwicker [7], can be simulated efficiently within \( \epsilon \)-error in the Robertson-Webb model when the measures of the players are bounded.
2 Model

The resource (cake) is represented as the interval \([0, 1]\). There is a set of players \(N = \{1, \ldots, n\}\) interested in the cake, such that each player \(i \in N\) is endowed with a private valuation function \(V_i\) that assigns a value to every subinterval of \([0, 1]\). These values are induced by a non-negative integrable value density function \(v_i\), so that for an interval \(I\), \(V_i(I) = \int_{x \in I} v_i(x) \, dx\). The value densities are non-atomic, so sets of measure zero are worth zero to every players. Without loss of generality, the valuations are normalized to \(V_i([0, 1]) = 1\), \(\forall i \in N\).

A piece of cake is a finite union of disjoint intervals. A piece is connected if it consists of a single interval. An allocation \(A = (A_1, \ldots, A_n)\) is a partition of the cake among the players, such that each player \(i\) receives the piece \(A_i\), the pieces are disjoint, and \(\bigcup_{i \in N} A_i = [0, 1]\).

An allocation \(A\) is said to be proportional if \(V_i(A_i) \geq 1/n\) for all \(i \in N\); envy-free if \(V_i(A_i) \geq V_i(A_j)\) for all \(i, j \in N\); perfect if \(V_i(A_j) = 1/n\) for all \(i, j \in N\); equitable if \(V_i(A_i) = c\), for all \(i \in N\) and some value \(c \in [0, 1]\). In addition to allocations, we will refer to partitions as divisions of the cake into pieces where the number of pieces need not equal the number of players. A partition \(A = (A_1, \ldots, A_k)\) is said to be a \(k\)-measure splitting if \(V_i(A_j) = 1/k\) for each player \(i\) and piece \(A_j\).

Envy-free allocations with connected pieces and equitable allocations with connected pieces always exist [34, 12], while \(k\)-measure splittings exist with \(n(k - 1)\) cuts [1].

We will also be interested in \(\epsilon\)-fair division, where the fairness constraints are satisfied within \(\epsilon\)-error; for instance, an allocation \(A\) is \(\epsilon\)-envy-free if \(V_i(A_i) \geq V_i(A_j) - \epsilon\), for each \(i, j = 1, \ldots, n\), and an \((\epsilon, k)\) measure splitting if \(V_i(A_j) \in (1/k - \epsilon, 1/k + \epsilon)\) for each \(i = 1, \ldots, n\), \(j = 1, \ldots, k\).

2.1 Query Complexity

All the discrete cake cutting protocols operate in a query model known as the Robertson-Webb model (see, e.g., the book by Robertson and Webb [27]), which was explicitly stated by Woeginger and Sgall [37]. In this model, the protocol communicates with the players using the following types of queries:

- **Cut\(_i\)(\(\alpha\))**: Player \(i\) cuts the cake at a point \(y\) where \(V_i([0, y]) = \alpha\). The point \(y\) becomes a cut point.

- **Eval\(_i\)(\(y\))**: Player \(i\) returns \(V_i([0, y])\), where \(y\) is a previously made cut point.

Note that the value of a piece \([x, y]\) can be determined with two Eval queries, Eval\(_i\)(\(x\)) and Eval\(_i\)(\(y\)).

A second class of protocols is known as “moving-knife” (or continuous) procedures, which typically involve sliding multiple knives across the cake, while evaluating the players’ valuations until some stopping condition is met. This class has not been formalized until now. Examples of such procedures include
Austin’s procedure, which computes a perfect allocation for two players, Stromquist’s procedure, for finding a connected envy-free allocation for three players, and Dubins-Spanier, for computing a proportional allocation for any number of players. The latter is the only (known) moving knife procedure that can be simulated exactly in the Robertson-Webb model.

The known bounds (both from previous literature and here) on the number of queries required to compute fair allocations are summarized in Table 2.1.

| Fairness notion       | Number of players | Upper bound | Lower bound |
|-----------------------|-------------------|-------------|-------------|
| \(\varepsilon\)-envy-free (connected) | \(n = 2\) | \(O(\log \varepsilon^{-1})\) (*) | \(\Omega(\log \varepsilon^{-1})\) (*) |
|                       | \(n = 3\) | \(O(n/\varepsilon)\) (*) | \(\Omega(\log \varepsilon^{-1})\) (*) |
|                       | \(n \geq 4\) | \(O(n/\varepsilon)\) (*) | \(\Omega(\log \varepsilon^{-1})\) (*) |
| \(\varepsilon\)-perfect (minimum cuts) | \(n = 2\) | \(O(\log \varepsilon^{-1})\) (*) | \(\Omega(\log \varepsilon^{-1})\) (*) |
|                       | \(n \geq 3\) | \(O\left(\frac{n^2}{\varepsilon}\right)\) [11] | \(\Omega\left(\frac{\log \varepsilon^{-1}}{\log \log \varepsilon^{-1}}\right)\) [26] |
| \(\varepsilon\)-equitable (connected) | \(n = 2\) | \(O(\log \varepsilon^{-1})\) [13] | \(\Omega(\log \varepsilon^{-1})\) (*) |
|                       | \(n \geq 3\) | \(O\left(n \left(\log n + \log \varepsilon^{-1}\right)\right)\) [13] | \(\Omega\left(\frac{\log \varepsilon^{-1}}{\log \log \varepsilon^{-1}}\right)\) [26] |
| envy-free (exact)     | \(n \geq 2\) | \(O\left(n^{\frac{n}{n-1}}\right)\) [3] | \(\Omega(n^2)\) [24] |
| proportional (exact)  | \(n \geq 2\) | \(O\left(n \log n\right)\) [19] | \(\Omega(n \log n)\) [37, 18] |

Table 1: Query complexity in cake cutting in the Robertson-Webb model. Our results are marked with (*). The lower bounds for finding \(\varepsilon\)-perfect and \(\varepsilon\)-equitable allocations for \(n \geq 3\) players hold for any number of cuts [26]. The bounds for exact envy-free and proportional allocations hold for any number of cuts, except the upper bound for proportional works for connected pieces.

3 General Simulation of Partitions

A general technique useful for computing approximately fair allocations in the Robertson-Webb model is based on asking the players to submit a discretization of the cake in many small cells via Cut queries, and reassembling them offline in a way that satisfies approximately the desired solution.

**Lemma 1.** Consider a cake cutting problem with \(n\) players. Given any partition of the cake \(A = (A_1, \ldots, A_m)\) in \(m\) pieces with at most cuts \(K\), then for all \(\varepsilon > 0\) there is a Robertson-Webb algorithm that asks \(O(K/\varepsilon)\) queries and computes a partition \(\bar{A} = (\bar{A}_1, \ldots, \bar{A}_m)\) with at most \(K\) cuts such that \(V_i(A_j) - \varepsilon \leq V_i(\bar{A}_j) \leq V_i(A_j) + \varepsilon\), for all \(i = 1 \ldots n, j = 1 \ldots m\).

**Proof.** Given a cake cutting instance, let \(x = (x_1, \ldots, x_K)\) be the cut points demarcating the partition \(A = (A_1, \ldots, A_m)\). Then consider the following Robertson-Webb protocol:
• Ask each player \( i \) a number of \( B = \lceil 2K/\epsilon \rceil \) Cut queries so that the cake is partitioned in \( B \) intervals each worth \( 1/B \) to \( i \).

• For each partition \( \tilde{A} = (\tilde{A}_1, \ldots, \tilde{A}_m) \) that can be attained by assembling the resulting \( n \cdot B \) cells in \( m \) pieces: if \( V_i(\tilde{A}_j) \in [V_i(A_j) - \epsilon, V_i(A_j) + \epsilon] \), output \( \tilde{A} \) and exit.

To see that existence of such a partition \( \tilde{A} \) is guaranteed, note that by rounding the cuts \( x_1 \ldots x_K \) of the exact partition \( A \) to the nearest point on the grid submitted by the players and allocating from left to right the resulting intervals in the same order as in \( A \) (including empty pieces if rounded cuts overlap), we obtain a partition \( \tilde{A} \) in which every piece is composed of at most \( K \) disconnected intervals. The error at each endpoint of an interval is at most \( 1/B \), thus the overall loss for any piece \( A_j \) from the point of view of any player \( i \) is bounded by \( 2K/B \leq \epsilon \), and so \( V_i(\tilde{A}_j) \in [V_i(A_j) - \epsilon, V_i(A_j) + \epsilon] \). \( \square \)

The lemma implies that it is possible to approximate in the RW model solutions that exist with a bounded number of cuts and can be expressed as a set of linear constraints. The difference from the work of [4], where discretizations are used to simulate the outcomes of bounded algorithms, is that here we do not restrict ourselves to properties that are computable by a discrete (Robertson-Webb) algorithm. In fact in our case a solution need not even exist on all instances. Before stating the theorem, given a cake cutting problem with \( n \) players and a partition \( A = (A_1, \ldots, A_m) \), denote the vector containing the valuation of each player \( i \) for every piece \( A_j \) by \( v(A) = (V_1(A_1), \ldots, V_1(A_m), \ldots, V_n(A_m)) \).

Theorem 3.1. Consider a cake cutting problem with \( n \) players. Let there be matrices \( c \in \mathbb{R}^{\ell \times mn} \) and \( b \in \mathbb{R}^{\ell \times 1} \), where \( c \) is normalized so that \( \sum_{j=1}^{mn} |c_{t,j}| \leq 1 \) for each row \( t = 1 \ldots \ell \). If there exists a partition \( A = (A_1, \ldots, A_m) \) with at most \( K \) cuts for which \( c \cdot v(A) \leq b \), then for each \( \epsilon > 0 \) there is a Robertson-Webb algorithm that asks \( O(K/\epsilon) \) queries and computes a partition \( \tilde{A} = (\tilde{A}_1, \ldots, \tilde{A}_m) \) that satisfies the linear constraints approximately; that is, \( c \cdot v(\tilde{A}) \leq b + \epsilon \), for all \( i = 1 \ldots n, j = 1 \ldots m \).

Proof. Let \( x_1 \ldots x_K \) be the cut points of a partition \( A = (A_1, \ldots, A_m) \) that satisfies the constraints \( c \cdot v(A) \leq b \). Consider any row \( t \) of \( c \). Then \( \sum_{j=1}^{mn} c_{t,j} \cdot v(A)_j \leq b_t \). Since \( c \) is normalized so that \( \sum_{j=1}^{mn} |c_{t,j}| \leq 1 \), we get \( c_{t,j} \in [-1,1] \) for all \( t, j \). By Lemma 1, we can find using \( O(K/\epsilon) \) queries a partition \( \tilde{A} \) s.t. \( V_i(\tilde{A}_j) - \epsilon \leq V_i(\tilde{A}_j) \leq V_i(\tilde{A}_j) + \epsilon \) for all \( i = 1 \ldots n, j = 1 \ldots m \). Thus \( c_{t,j} \cdot v(\tilde{A}_j) \leq c_{t,j} \cdot v(A)_j + \epsilon \cdot |c_{t,j}| \). Summing over all \( j \), we get \( \sum_{j=1}^{mn} c_{t,j} \cdot v(\tilde{A})_j \leq \sum_{j=1}^{mn} c_{t,j} \cdot v(A)_j + \epsilon \cdot \sum_{j=1}^{mn} |c_{t,j}| = \sum_{j=1}^{mn} c_{t,j} \cdot v(A)_j + \epsilon \cdot \sum_{j=1}^{mn} |c_{t,j}| \leq b_t + \epsilon \). Thus partition \( \tilde{A} \) is a solution, which completes the proof. \( \square \)

As a corollary, we get the following bounds for computing approximately fair allocations in the Robertson-Webb model. This also implies a result in [11], which considered discretizations in the context of computing \( \epsilon \)-perfect allocations.

Corollary 1. For each \( \epsilon > 0 \) and number of players \( n \in \mathbb{N} \), a partition that is

• \( \epsilon \)-envy-free and connected can be computed with \( O(n/\epsilon) \) queries
• \(\epsilon\)-equitable, proportional, and connected can be computed with \(O(n/\epsilon)\) queries
• an \((\epsilon, k)\)-measure splitting with \(k\) pieces can be computed with \(O(n(k - 1)/\epsilon)\) queries.

4 Envy-Free Allocations

Exact connected envy-free allocations are guaranteed to exist (see, e.g., Stromquist [32], Su [34]), but cannot be computed by finite RW protocols (Stromquist [33]). However, the impossibility result of Stromquist does not explain how many queries are needed to obtain connected \(\epsilon\)-envy-free allocations.

From Corollary 1, a connected \(\epsilon\)-envy-free allocation can be computed with \(O(n/\epsilon)\) queries. As we show next, fewer queries are needed for three players. The proof relies on approximately computing the outcome of a moving knife procedure due to Barbanel and Brams [5].

Barbanel-Brams procedure: Ask each player \(i\) to return the point such that one third of the cake is to the right of it. If an envy-free allocation can be formed with the pieces demarcated by the player who had the rightmost mark (say 1)—\([0, \ell], [\ell, r], [r, 1]\)—output it. Otherwise there are two cases:

Case 1: Both players 2 and 3 strictly prefer the piece \([\ell, r]\). Then move a sword continuously from \(r\) towards \(\ell\), keeping for each position \(z\) of the sword the point \(t\) for which \(V_1(0, t) = V_1(z, 1)\). By the intermediate value theorem, there exists position of the sword such that one of 2, 3 is indifferent between two pieces, and an envy-free allocation exists at cuts \(t, z\).

Case 2: Both players 2 and 3 strictly prefer the piece \([0, \ell]\). Then move a sword continuously from \(\ell\) towards 0, keeping for each position \(z\) of the sword the point \(t\) such that \(V_1(z, t) = V_1(t, 1)\). By the intermediate value theorem, there exists position of the sword such that one of the players 2 and 3 is indifferent between the piece \([0, z]\) and one of \([z, t], [t, 1]\), which yields an envy-free allocation with cuts \(t, z\).

Theorem 4.1. A connected \(\epsilon\)-envy-free allocation for three players can be computed with \(O(\log \frac{1}{\epsilon})\) queries.

Proof. We will compute an \(\epsilon\)-envy-free allocation using several steps, not all of which are not included in the Barbanel-Brams protocol. The first step of the Barbanel-Brams protocol is discrete, and so it can be executed with \(O(1)\) Robertson-Webb queries.

If the instance falls in Case 1, we will maintaining the next invariant:

(a) there exist points \(0 < w < z < 1\), such that there for some points \(a < b, a, b \in [0, w)\) we have \(V_1(0, a) = V_1(z, 1), V_1(0, b) = V_1(w, 1)\), and players 2 and 3 agree that \((i)\) the piece \([a, z]\) is larger than any of \([0, a]\) and \([z, 1]\) by more than \(\epsilon\), while \((ii)\) the piece \([b, w]\) is smaller than one of \([0, b]\) and \([w, 1]\) by more than \(\epsilon\).
Consider the following procedure:

1. Initialize \( w \) and \( z \) by setting \( z = r \) and asking player 1 a Cut query to identify its midpoint \( w \) of the cake. Clearly the allocation made with cut points \(( \ell, r )\) has the property that both players 2 and 3 prefer the middle piece, \([ \ell, r ]\), while if the cuts overlap on \( w \), then both 2 and 3 prefer one of the pieces \([0, w] \) or \([w, 1]\).

2. Given points \( w, z \) that satisfy invariant \((a)\), for which \( V_1(w, z) \geq \epsilon \), ask player 1 iteratively a Cut query to determine the midpoint \( m \in [w, z] \), for which \( V_1(w, m) = V_1(m, z) \). Then ask player 1 a Cut query as to return the point \( m' \) for which \( V_1(0, m') = V_1(m, 1) \). If there is an \( \epsilon \)-envy-free allocation with cuts \( m' \) and \( m \), output it. Otherwise, if both players 2 and 3 evaluate the piece \([m', m] \) as the largest among \{ \([0, m']\), \([m', m]\), \([m, 1]\) \}, then set \( z = m \). Else, it must be the case that both 2 and 3 estimate the piece \([m', m] \) as strictly smaller than at least one of \([0, m']\) and \([m, 1]\) by more than \( \epsilon \); set \( w = m \). Note the new points \( w, z \) still satisfy property \((a)\).

Figure 1: Case 1 of the simulation.

Step 1 requires a constant number of queries, while Step 2 is executed at most \( \log \frac{1}{\epsilon} \) times, since the valuation of player 1 for the interval \([w, z]\) halves with each round; each round of Step 2 requires a constant number of queries. If an \( \epsilon \)-envy-free allocation has been found after completing Steps 1 and 2, the case is complete. Otherwise, we will further reduce the interval \([w, z]\) until it becomes small also from the point of view of player 2. Note that the invariant still holds after completing the previous steps, so there exist \( a < b < w \) such that \( V_1(w, z) = V_1(a, b) = \delta < \epsilon \). By the intermediate value theorem, we are guaranteed to have an \( \epsilon \) envy-free solution with cuts \( x_1 \in [a, b] \) and \( x_2 \in [w, z] \). Moreover, since \( V_1([a, b]) \leq 1/3 \), any allocation obtained with cuts \( x_1 \in [a, b] \), \( x_2 \in [w, z] \) such that player 1 receives one of the pieces \([0, x_1]\), \([x_2, 1]\) is \( \epsilon \)-envy-free for player 1.

Recall the piece \([a, z]\) is strictly larger than \( \epsilon \) than \([0, a]\) and \([z, 1]\) in the estimation of players 2 and 3. We have three subcases, depending on the piece \([b, z]\).

- Exactly one of players 2 and 3 views \([b, z]\) as the largest piece among \([0, b]\), \([b, z]\), and \([z, 1]\). Then an \( \epsilon \)-envy-free allocation can be obtained with cuts \( b \) and \( z \).
- Both players 2 and 3 view \([b, z]\) as larger than \([0, b]\) and \([z, 1]\). Then there must exist an \( \epsilon \)-envy-free solution with cuts \( b \) and \( x_2 \in [w, z] \). Such a solution can be found with binary search on \([w, z]\), using

\[8\]
the valuation of player 2 to half the interval \([w, z]\) in each iteration. The solution \(x_2\) reached this way will have the property that player 2 is indifferent (within \(\epsilon\)) between \([b, x_2]\) and one of the outside pieces \([0, b]\) or \([x_2, 1]\).

- Otherwise, both players 2 and 3 view the piece \([b, z]\) as smaller than one of \([0, b]\) and \([z, 1]\). Then by the intermediate value theorem there exists an envy-free allocation with cuts \(x_1 \in [a, b]\) and \(z\). We can find an approximate solution with binary search on the interval \([a, b]\) using the valuation of player 2 to repeatedly identify the midpoint of \([a, b]\).

Each subcase completes with \(O(\log \frac{1}{\epsilon})\) queries, which gives an overall bound of \(O(\log \frac{1}{\epsilon})\) for Case 1.

Otherwise, we enter Case 2, for which we maintain the invariant:

(b) there exist points \(0 < w < z < 1\), such that for some points \(a, b\) with \(w < a < b \leq 1\) we have \(V_1(w, a) = V_1(a, 1)\), \(V_1(z, b) = V_1(b, z)\), and players 2 and 3 agree that (i) the piece \([0, z]\) is larger than any of \([z, b]\) and \([b, 1]\) by more than \(\epsilon\), while (ii) the piece \([0, w]\) is smaller than one of \([w, a]\) and \([a, 1]\) by more than \(\epsilon\).

The steps for simulating Case 2 are:

3. Initialize \(w = 0\) and \(z = \ell\). Ask player 1 a Cut query to identify the midpoint \(q\) of the cake in its estimation. An allocation made with cuts \(\ell\) and \(r\) has the property that both players 2 and 3 view the leftmost piece \([0, \ell]\) as the largest, while the allocation made with pieces \([0, 0], [0, q], [q, 1]\) has the property that none of the players 2 and 3 want the (now empty) leftmost piece.

4. Given points \(w, z\) that satisfy invariant \(a\), for which \(V_1(w, z) \geq \epsilon\), iteratively ask player 1 a Cut query to determine the midpoint \(m \in [w, z]\) for which \(V_1(w, m) = V_1(m, z)\). Then find, via another Cut query, the point \(m' \in [z, 1]\) for which \(V_1(z, m') = V_1(m', 1)\). If an \(\epsilon\)-envy-free allocation exists with cuts \(m\) and \(m'\), output it. Otherwise, if both players 2 and 3 strictly prefer piece \([0, m]\) to any of \([m, m']\) and \([m', 1]\), then update \(z = m\). Else, both players 2 and 3 view the piece \([0, m]\) as strictly smaller than at least one of \([m, m']\) and \([m', 1]\) by more than \(\epsilon\); set \(w = m\).

![Figure 2](image-url)
Step 3 requires a constant number of queries, while Step 4 at most $\log_{\epsilon} z$ queries. If an $\epsilon$-envy-free allocation has not been found after completing steps 3-4, then since the interval $[w, z]$ is worth less than $\epsilon$ to player 1, we can again reduce the problem to finding an agreement among players 2 and 3, as was done in Case 1. This completes the proof.

Next we prove this upper bound is tight by generalizing a construction of Stromquist [33], who showed that no finite RW protocol can find an exact connected envy-free allocation of the cake. Towards this end, the family of “rigid measure systems” was introduced, in which the valuations are consistent with a circulant matrix for a choice of the cut points. We first generalize this family of valuations, then show how it can be used to derive a lower bound for the problem of computing $\epsilon$-envy-free connected allocations.

**Definition 1** (Generalized Rigid Measure System). A tuple of value density functions $V = (v_1, v_2, v_3)$ is a generalized rigid measure system if the following conditions are met:

- the density of each measure is bounded: $\frac{1}{\sqrt{2}} < v_i(x) < \sqrt{2}$, for all $x \in [0, 1]$.
- there exist points $x, y \in [0, 1]$ and values $t_i, s_i$ for each player $i$ such that $0 < s_i < 1/3 < t_i < 1/2$ and the matrix of valuations for the pieces demarcated by these points is given by the following table.

|      | $[0, x]$ | $[x, y]$ | $[y, 1]$ |
|------|----------|----------|----------|
| $V_1$| $t_1$    | $t_1$    | $s_1$    |
| $V_2$| $s_2$    | $t_2$    | $t_2$    |
| $V_3$| $t_3$    | $s_3$    | $t_3$    |

Table 2: Generalized Rigid Measure System: 3 players

Generalized rigid measure systems satisfy the property that the valuations of the players for any given piece cannot differ too much.

**Lemma 2.** Consider any cake cutting problem where for two players $i$ and $j$ there exist $a, b > 0$ such that for all $x \in [0, 1]$, $1/a < v_i(x) < b$ and $1/a < v_j(x) < b$. Then for any two pieces $S_1, S_2$ of the cake, if $V_i(S_1) \geq ab \cdot V_i(S_2)$, it follows that $ab \cdot V_j(S_1) > V_j(S_2)$.

**Proof.** This generalizes a lemma given by Stromquist [33]. Let $S_1$ and $S_2$ be two pieces of lengths $\ell_1$ and $\ell_2$, respectively, such that $V_i(S_1) \geq ab \cdot V_i(S_2)$. By using the constraints on the densities, we get:

$$ab \cdot V_j(S_1) > ab \cdot \left(\frac{\ell_1}{a}\right) = b \cdot \ell_1 > V_i(S_1) \geq ab \cdot V_i(S_2) > ab \cdot \left(\frac{\ell_2}{a}\right) = b \cdot \ell_2 > V_j(S_2)$$

This completes the proof. \qed
A useful notion to measure how close a protocol is to discovering a generalized rigid measure system on a given instance is that of a partial rigid measure system.

**Definition 2** (Partial Rigid Measure System). A tuple of valuations $V = (v_1, v_2, v_3)$ is a partial rigid measure system if

- the density of each player $i$ is bounded everywhere: $\frac{1}{\sqrt{2}} < v_i(x) < \sqrt{2}$, for all $x \in [0, 1]$.
- There exist values $k > 0$ and $1/2 > \ell_i > 1/3 > m_i > 0$ for each player $i$, and points $x, x', y, y' \in [0, 1]$, such that the matrix of valuations for pieces demarcated by these points is given by the following table.

|        | $[0, x]$ | $[x, x']$ | $[x', y]$ | $[y, y']$ | $[y', 1]$ |
|--------|----------|-----------|-----------|-----------|-----------|
| $V_1$  | $\ell_1$ | $k$       | $\ell_1$  | $k$       | $m_1$     |
| $V_2$  | $m_2$    | $k$       | $\ell_2$  | $\ell_2$  |           |
| $V_3$  | $\ell_3$ | $k$       | $m_3$     | $k$       | $\ell_3$  |

Table 3: Partial Rigid Measure System: 3 players

Partial rigid measure systems have the property that if there is a collection of cut points $\mathcal{P} = \{z_1, \ldots, z_k\} \subset [0, 1]$, such that there are no points from $\mathcal{P}$ in the intervals $(x, x')$ and $(y, y')$, then any connected partition attainable using cut points from $\mathcal{P}$ has envy at least $0.01k$.

**Lemma 3.** Let $V = (v_1, v_2, v_3)$ be a partial rigid measure system and $\mathcal{P} = \{z_1, \ldots, z_k\} \subset [0, 1]$ a collection of cut points such that $x, x', y, y' \in \mathcal{P}$ and there are no points from $\mathcal{P}$ in the intervals $(x, x')$ and $(y, y')$. Then any connected partition demarcated by points in $\mathcal{P}$ has envy at least $0.01k$.

**Proof.** Suppose by contradiction that there exists an allocation with cut points $\hat{x}, \hat{y} \in \mathcal{P}$ so that each player is not envious by more than $0.01k$. Recall that $\hat{x}, \hat{y} \notin \{(x, x'), (y, y')\}$. We have a few cases:

**Case 1:** $\hat{x} \leq x$. Regardless of who owns the piece $[0, \hat{x}]$, it cannot be the case that $\hat{y} \leq x$, since that player would envy the right remainder of the cake, namely the interval $[\hat{y}, 1]$, by an amount much larger than $k$. We consider several scenarios, depending on the owner of $[0, \hat{x}]$.

**Case 1.a** Player 1 receives $[0, \hat{x}]$. There are two subcases:

- $\hat{y} \in [x', y]$. Let $S_1 = [\hat{x}, x]$ and $S_2 = [\hat{y}, y]$. Since player 1 does not envy any other piece by more than $0.01k$, we have

  $V_1(0, \hat{x}) \geq V_1(\hat{x}, \hat{y}) - 0.01k \iff \ell_1 - V_1(0, x) = V_1(0, x) - V_1(S_1)$

  $\geq V_1(x, y) + V_1(S_1) - V_1(S_2) - 0.01k = \ell_1 + k + V_1(S_1) - V_1(S_2) - 0.01k$

  $\iff V_1(S_2) \geq 2(V_1(S_1) + 0.495k)$
By Lemma 2, we have that $2V_j(S_2) > V_j(S_1) + 0.495k$ for all $j \in \{2, 3\}$. Then both players 2 and 3 will prefer the rightmost piece, $[\hat{y}, 1]$, by a margin of at least $0.01k$. For player 2, we have

$$V_2(\hat{y}, 1) - V_2(\hat{x}, \hat{y}) = V_2(\hat{y}, y) + k + V_2(y', 1) - (V_2(x, y) + V_2(S_1) - V_2(S_2))$$

$$= k + V_2(S_2) + \ell_2 - (\ell_2 + k + V_2(S_1) - V_2(S_2)) = 2V_2(S_2) - V_2(S_1)$$

$$> 0.495k > 0.01k$$

For player 3, we have

$$V_3(\hat{y}, 1) - V_3(\hat{x}, \hat{y}) = V_3(\hat{y}, y) + k + V_3(y', 1) - (V_3(x, y) + V_3(S_1) - V_3(S_2))$$

$$= 2V_3(S_2) - V_3(S_1) + (\ell_3 - m_3) > 0.495k + \ell_3 - m_3 > 0.01k$$

- $\hat{y} \geq y'$. Then player 1 will envy the middle piece by at least $2k$:

$$V_1(\hat{x}, \hat{y}) - V_1(0, \hat{x}) = V_1(\hat{x}, x) + k + \ell_1 + k + V_1(y', \hat{y}) - V_1(0, \hat{x}) \geq \ell_1 + 2k - \ell_1 = 2k$$

**Case 1.b)** Player 2 receives $[0, \hat{x}]$. Then by approximate envy-freeness, we have

$$V_2(0, \hat{x}) \geq V_2(\hat{x}, \hat{y}) - 0.01k \text{ and } V_2(0, \hat{x}) \geq V_2(\hat{y}, 1) - 0.01k$$

By summing up the two inequalities, we get

$$V_2(0, \hat{x}) \geq 2V_2(\hat{x}, \hat{y})/2 - 0.01k = (1 - V_2(0, \hat{x}))/2 - 0.01k \iff 3V_2(0, \hat{x})/2 \geq 1/2 - 0.01k$$

$$\iff V_2(0, \hat{x}) \geq 1/3 - 0.02k/3$$

Since $m_2 \geq V_2(0, \hat{x})$, $2\ell_2 + m_2 + 2k = 1$, and $\ell_2 > 1/3$, it we have $m_2 = 1 - 2\ell_2 - 2k < 1/3 - 2k$, thus $1/3 - 2k > m_2 \geq V_2(0, \hat{x}) \geq 1/3 - 0.02k/3$. This is a contradiction, so the case cannot happen.

**Case 1.c)** Player 3 receives $[0, \hat{x}]$. We have the following subcases:

- $\hat{y} \in [x', y]$. Then $V_3(0, \hat{x}) \leq \ell_3$, while $V_3(\hat{y}, 1) \geq \ell_3 + k$, thus player 3 envies the piece $[\hat{y}, 1]$ by more than $0.01k$.

- $\hat{y} \geq y'$. Then both players 1 and 2 value the middle piece $[\hat{x}, \hat{y}]$ more than the rightmost piece $[\hat{y}, 1]$, and the difference is larger than $0.01k$. For player 2 we have $V_2(\hat{x}, \hat{y}) \geq \ell_2 + 2k$, while $V_2(\hat{y}, 1) \leq \ell_2$, and for player 1 we have $V_1(\hat{x}, \hat{y}) \geq \ell_1 + 2k$, while $V_1(\hat{y}, 1) \leq m_1 < \ell_1$.

Thus no player can accept the leftmost piece, $[0, \hat{x}]$, which completes Case 1.

**Case 2:** $\hat{x} \in [x', y]$. Then $\hat{y} \geq y'$, since no player would accept (even within envy $0.01k$) a piece smaller than $[x', y]$. Moreover, players 1 and 3 would not accept the piece $[\hat{y}, 1]$ since they would envy the player
owning \([0, \hat{x}]\) by more than \(k\). Thus the piece \([\hat{y}, 1]\) can only be assigned to player 2. Let \(S_1 = [x', \hat{x}]\) and \(S_2 = [y', \hat{y}]\). By approximate envy-freeness of player 2, we have

\[
V_2(\hat{y}, 1) \geq V_2(\hat{x}, \hat{y}) - 0.01k = V_2(x', y) - V_2(S_1) + V_2(S_2) + k - 0.01k = \ell_2 - V_2(S_1) + V_2(S_2) + 0.99k
\]

Since \(V_2(\hat{y}, 1) = \ell_2 - V_2(S_2)\), we get that

\[
\ell_2 - V_2(S_2) \geq \ell_2 - V_2(S_1) + V_2(S_2) + 0.99k \iff V_2(S_1) \geq 2(V_2(S_2) + 0.495k)
\]

By Lemma 2, we have \(2V_j(S_1) \geq V_j(S_2) + 0.99k\) for all \(j \in \{1, 3\}\).

We wish to show that both players 1 and 3 would only accept the leftmost piece \([0, \hat{x}]\) from the remaining pieces. For player 1, we have

\[
V_1(0, \hat{x}) - V_1(\hat{x}, \hat{y}) = V_1(0, x) + V_1(x, x') + V_1(S_1) - (V_1(x', y) - V_1(S_1) + k + V_1(S_2))
\]

\[
= \ell_1 + k + V_1(S_1) - \ell_1 + V_1(S_1) - k - V_1(S_2) = 2V_1(S_1) - V_1(S_2)
\]

\[
\geq 0.495k > 0.01k
\]

For player 3, using the fact that \(\ell_3 > m_3\), we have

\[
V_3(0, \hat{x}) - V_3(\hat{x}, \hat{y}) = V_3(0, x) + V_3(x, x') + V_3(S_1) - (V_3(x', y) - V_3(S_1) + k + V_3(S_2))
\]

\[
= \ell_3 + k + V_3(S_1) - m_3 + V_3(S_1) - k - V_3(S_2) = 2V_3(S_1) - V_3(S_2) + \ell_3 - m_3
\]

\[
\geq 0.495k + \ell_3 - m_3 > 0.01k
\]

**Case 3:** \(\hat{x} \geq y'\). This scenario is clearly infeasible, since all the players would envy the owner of the piece \([0, \hat{x}]\) by at least \(2k\).

In all the cases we obtained a contradiction, so any partition attained with cut points from \(\mathcal{P}\) has envy at least \(0.01k\). This completes the proof. \(\square\)

Next we show that if at some point during the execution of a protocol \(\mathcal{A}\) the valuations and cut points discovered are consistent with a partial rigid measure system with two slivers \(I = [x, x']\) and \(J = [y, y']\) of uniform density and value \(k\) for all the players, then the next cut query can be answered so that the configuration remains a partial rigid measure system, where the new slivers \(I'\) and \(J'\) still have uniform density and value at least \(0.01k\). We consider three scenarios, depending on which player receives the question.

**Lemma 4.** Let \(\mathcal{A}\) be a Robertson-Webb protocol and suppose that at some point during its execution, the valuations and cuts discovered are consistent with a partial rigid measure system as given in Table 2, such that the intervals \(I = [x, x']\) and \(J = [y, y']\) have value \(k\) to all the players, uniform density, and no cut points inside.

Then if any player \(i\) receives a cut query inside interval \(I\), there exist answers so that the new configuration is still consistent with a partial rigid measure system, with new intervals \(I'\) and \(J'\) of uniform density for all the players, length \(0.01k\), and with no cut points inside \(I'\) and \(J'\).
Proof. (part a) We show the proof for the scenario where the cut query is addressed to player 1. The analysis for the cases where players 2 or 3 receive the question can be found in the appendix.

Let \( P \) be the collection of cut points discovered by \( A \), where all the values of pieces induced by points in \( P \) and the endpoints of the cake are known. Since the configuration is consistent with a partial rigid measure system, let the values be denoted as in Table 2.

Denote by \( \text{Cut}(1; \alpha) \) the next cut query addressed to player 1. As stated in the lemma, we deal with the range of \( \alpha \in (\ell_1, \ell_1 + k) \). Since the density is uniform on \( I \) and \( J \), we have that \( x' = x + 0.01k \) and \( y' = y + 0.01k \). Consider two cases:

**Case 1:** \( \alpha \in (\ell_1, \ell_1 + 2k/3) \). We will hide the new interval \( I' \) on the right side of the point \( x + 2k/3 \). Define \( m = x + 2k/3 \) and \( n = x + 2k/3 + 0.01k \). Set \( I' = [m, n] \). Similarly on the right hand side, define \( p = y + 1.03k/3 \) and \( q = y + 1.03k/3 + 0.01k \). Set \( J' = [p, q] \). Update the collection of cut points to \( P' = P \cup \{m, n, p, q\} \). Let the value of each player \( i \) for the intervals \( I' \) and \( J' \) be exactly 0.01k. Since \( I' \) and \( J' \) have length 0.01k, the densities remain uniform in these intervals. Set the values of player 1 for the other new intervals to \( V_1(x, m) = 2k/3 \), \( V_1(n, x + k) = 0.97k/3 \), \( V_1(y, p) = 1.03k/3 \), and \( V_1(q, y + k) = 1.94k/3 \).

Denote the values of player 2 for the unknown intervals by \( V_2(x, m) = 0.99k - \lambda k \), \( V_2(n, x + k) = \lambda k \), \( V_2(y, p) = \mu k \), \( V_2(q, y + k) = 0.99k - \mu k \), where \( 0 < \lambda, \mu < 0.99 \). Clearly the values add up to the weight of \( I \) and \( J \) for player 2:

- \( V_2(x, x + k) = V_2(x, m) + V_2(m, n) + V_2(n, x + k) = 0.99k - \lambda k + 0.01k + \lambda k = k \)
- \( V_2(y, y + k) = V_2(y, p) + V_2(p, q) + V_2(q, y + k) = \mu k + 0.01k + 0.99k - \mu k = k \)

For player 3, the values of the unknown intervals are \( V_3(x, m) = 0.99k - \phi k \), \( V_3(n, x + k) = \phi k \), \( V_3(y, p) = \psi k \), and \( V_3(q, y + k) = 0.99k - \psi k \), where \( 0 < \phi, \psi < 0.99k \). The weights add up to the value of player 3 for the whole intervals \( I \) and \( J \); the check is similar to that of player 2 and thus omitted.

We obtain the following table of values, where the parameters \( \lambda, \mu, \phi, \psi \) must be determined.

|          | \([0, x]\) | \([x, m]\) | \([m, n]\) | \([n, x + k]\) | \([x + k, y]\) | \([y, p]\) | \([p, q]\) | \([q, y + k]\) | \([y + k, 1]\) |
|----------|------------|------------|------------|---------------|---------------|-------------|-------------|---------------|-------------|
| \(V_1\)  | \(\ell_1\) | \(2k/3\)   | \(0.01k\)  | \(0.97k/3\)   | \(\ell_1\)    | \(1.03k/3\)  | \(0.01k\)   | \(1.94k/3\)  | \(m_1\)     |
| \(V_2\)  | \(m_2\)    | \(0.99k - \lambda k\) | \(0.01k\)  | \(\lambda k\) | \(\ell_2\)    | \(\mu k\)    | \(0.01k\)   | \(0.99k - \mu k\) | \(\ell_2\)  |
| \(V_3\)  | \(\ell_3\) | \(0.99k - \phi k\) | \(0.01k\)  | \(\phi k\)    | \(m_3\)       | \(\psi k\)    | \(0.01k\)   | \(0.99k - \psi k\) | \(\ell_3\)  |

Table 4: Partial Rigid Measure System for \( \alpha \in (\ell_1, \ell_1 + 2k/3) \). The break points are \( m = x + 2k/3 \), \( n = x + 2k/3 + 0.01k \), \( p = y + 1.03k/3 \), \( q = y + 1.03k/3 + 0.01k \). Densities are uniform on \([m, n], [p, q]\).
The remaining requirements for the new configuration to form a partial rigid measure system are that the values yield a new configuration with parameters $\ell'_1, m'_1$ and all the densities on $[x, m], [n, x + k], [y, p],$ and $[q, y + k]$ are in the required bounds of $1/\sqrt{2}$ and $\sqrt{2}$.

For player 1, by choice of values we have that $\ell'_1 = \ell_1 + 2k/3 = 0.97k/3 + \ell_1 + 1.03k/3$ and $m'_1 = m_1 + 1.94k/3$. Moreover, it can be verified that player 1’s density will be uniform on all of the new intervals, which clearly belongs to the range $(1/\sqrt{2}, \sqrt{2})$. For player 2 we must find $0 < \lambda, \mu < 0.99$ such that the values still form a partial rigid measure system. Thus $\lambda k + \ell_2 + \mu k = 0.99k - \mu k + \ell_2 \iff \lambda + 2\mu = 0.99 \iff \mu = 0.495 - \lambda/2$. By choice of the points $m, n, p, q$ and the valuations, the densities of player 2 on the different intervals are:

- $[x, m] : V_2(x, m)/(m - x) = (0.99k - \lambda k)/(x + 2k/3 - x) = (297 - 300\lambda)/200$
- $[n, x + k] : V_2(n, x + k)/(x + k - n) = \lambda k/(k/3 - 0.01k) = 300\lambda/97$
- $[y, p] : V_2(y, p)/(p - y) = \mu k/(y + 1.03k/3 - y) = 297/206 - 150\lambda/103$
- $[q, y + k] : V_2(q, y + k)/(y + k - q) = (0.99k - \mu k)/(y + k - y - 1.03k/3 - 0.01k) = 297/388 + 150\lambda/194$

By setting, for instance, $\lambda = 0.25$, all the densities fall in $(1/\sqrt{2}, \sqrt{2})$ and there exists a solution for player 2. Then $\mu = 0.37, m'_2 = m_2 + 0.99k - 0.25k, and \ell'_2 = \lambda k + \ell_2 + \mu k = 99k/100 - \mu k + \ell_2$.

Finally, for player 3, we must find $0 < \phi, \psi < 0.99$ such that $\ell_3 + 0.99k - \phi k = 0.99k - \psi k + \ell_3 \iff \phi = \psi$. The densities of player 3 are:

- $[x, m] : V_3(x, m)/(m - x) = (0.99k - \phi k)/(x + 2k/3 - x) = 1.485 - 1.5\phi$
- $[n, x + k] : V_3(n, x + k)/(x + k - n) = \phi k/(k/3 - 0.01k) = 300\phi/97$
- $[y, p] : V_3(y, p)/(p - y) = \phi k/(y + 1.03k/3 - y) = 300\phi/103$
- $[q, y + k] : V_2(q, y + k)/(y + k - q) = (0.99k - \phi k)/(y + k - y - 1.03k/3 - 0.01k) = 297/194 - 300\phi/194$

By setting, for example, $\phi = \psi = 0.25$, we obtain the density of player 3 to be in $(1/\sqrt{2}, \sqrt{2})$ on all the intervals. Let $\ell'_3 = \ell_3 + 0.99k - 0.25k and m'_3 = m_3 + 0.5k$.

Finally we can answer the query. Since $\ell_1 < \alpha < \ell_1 + 2k/3$, and the values of all the players on the interval $[x, m]$ have been set, find the point $z$ with the property that $V_1(x, z) = \alpha$ and the density is uniform for player 1 on $[x, z]$. Then fit the answers for the other two players, proportional with their average density on $[x, z]$, and update $P'$ to include the point $z$. 

15
**Case 2:** \( \alpha \in (\ell_1 + 2k/3, \ell_1 + k) \). This time the interval \( I' \) will be hidden on the left side of \( x + 2k/3 \). Define \( m = x + 2k/3 - 0.01k \) and \( n = x + 2k/3 \). Set \( I' = [m, n] \). Let \( p = y + 0.97k/3 \) and \( q = y + k/3 \). Set \( J' = [p, q] \). Update the collection of cut points to \( \mathcal{P}' = \mathcal{P} \cup \{m, n, p, q\} \). Let the value of each player \( i \) for the intervals \( I' \) and \( J' \) be exactly \( 0.01k \); again the densities remain uniform on \( I' \) and \( J' \). Set the values of player 1 for the other intervals to \( V_1(x, m) = 2k/3 - 0.01k \), \( V_1(n, x + k) = k/3 \), \( V_1(y, p) = 0.97k/3 \), and \( V_1(q, y + k) = 2k/3 \). It can be verified that player 1’s density is uniform on all the new intervals. Update \( \ell_1' = \ell_1 + 2k/3 - 0.01k \) and \( m_1' = m_1 + 2k/3 \).

For players 2 and 3 we must find parameters \( 0 < \lambda, \mu, \phi, \psi < 0.99 \) such that the following matrix of valuations is compatible with a partial rigid measure system.

|       | \([0, x]\) | \([x, m]\) | \([m, n]\) | \([n, x + k]\) | \([x + k, y]\) | \([y, p]\) | \([p, q]\) | \([q, y + k]\) | \([y + k, 1]\) |
|-------|------------|------------|------------|--------------|--------------|----------|----------|--------------|--------------|
| \(V_1\) | \(\ell_1\) | 1.97k/3 | 0.01k | \(k/3\) | \(\ell_1\) | 0.97k/3 | 0.01k | \(2k/3\) | \(m_1\) |
| \(V_2\) | \(m_2\) | 0.99k - \(\lambda\)k | 0.01k | \(\lambda\)k | \(\ell_2\) | \(\mu\)k | 0.01k | 0.99k - \(\mu\)k | \(\ell_2\) |
| \(V_3\) | \(\ell_3\) | 0.99k - \(\phi\)k | 0.01k | \(\phi\)k | \(m_3\) | \(\psi\)k | 0.01k | 0.99k - \(\psi\)k | \(\ell_3\) |

Table 5: Partial Rigid Measure System for \( \alpha \in (\ell_1 + 2k/3, \ell_1 + k) \). The break points are \( m = x + 2k/3 - 0.01k \), \( n = x + 2k/3 \), \( p = y + 0.97k/3 \), \( q = y + k/3 \). All the densities are uniform on \([m, n]\) and \([p, q]\).

For player 2 we obtain from the analysis in Case 1 that \( \mu = 99/200 - \lambda/2 \). The densities are:

- \([x, m]\): \(V_2(x, m)/(m - x) = (0.99k - \lambda)k/(x + 2k/3 - 0.01k - x) = 297/197 - 300\lambda/197\)
- \([n, x + k]\): \(V_2(n, x + k)/(x + k - n) = \lambda k/(k/3) = 3\lambda\)
- \([y, p]\): \(V_2(y, p)/(p - y) = \mu k/(0.97k/3) = 300\mu/97\)
- \([q, y + k]\): \(V_2(q, y + k)/(y + k - q) = (0.99k - \mu k)/(2k/3) = 297/200 - 3\mu/2\)

Setting \( \lambda = 0.25 \) ensures the density on each of these intervals is in \((1/\sqrt{2}, \sqrt{2})\). Then \( \mu = 0.37 \). Update \( \ell_2' = \ell_2 + 0.62k \) and \( \mu_2' = m_2 + 0.74k \).

For player 3, by symmetry with Case 1, we have \( \phi = \psi \) and \( 0 < \phi < 0.99 \). The density of player 3 on the considered intervals is:

- \([x, m]\): \(V_3(x, m)/(m - x) = (0.99 - \phi)/(1.97/3) = 297/197 - 300\phi/197\)
- \([n, x + k]\): \(V_3(n, x + k)/(x + k - n) = \phi k/(k/3) = 3\phi\)
- \([y, p]\): \(V_3(y, p)/(p - y) = (\phi k)/(y + 1.03k/3 - y) = 300\phi/103\)
- \([q, y + k]\): \(V_3(q, y + k)/(y + k - q) = (0.99k - \phi k)/(2k/3) = 297/200 - 3\phi/2\)
Setting $\phi = \psi = 0.25$ ensures the density on each interval is in the $(1\sqrt{2}, \sqrt{2})$ range. Update $\ell_3' = \ell_3 + 0.74k$ and $m_3' = m_3 + 0.5k$.

We can now answer the query addressed to player 1. The interval $[0, m]$ has the property that $V_1(0, m) = \ell_1 + 2k/3$, and so $\alpha > V_1(0, m)$. Thus we can return a point $z \in (n, x + k)$ with the property that player 1’s density is uniform on $[n, z]$. Add $z$ to $P'$ and report the answers of the other players for the piece $[n, z]$ in a way that is proportional to their average density on $[n, x + k]$.

From the case analysis it follows that if player 1 receives a query falling inside interval $I$, we can find answers so that the new configuration is still a partial rigid measure system with the properties required by the lemma. Together with the analysis in part b and c (in appendix), this completes the proof of the lemma.

**Theorem 4.2. Computing a connected $\epsilon$-envy-free allocation for three players requires $\Omega \left( \log \frac{1}{\epsilon} \right)$ queries.**

**Proof.** Set the initial configuration to a partial rigid measure system as in the next table, where $k = 0.01$ and $\ell_i = 0.35$, $m_i = 0.28$ for each player $i$. The initial cuts are at $0.34, 0.35, 0.67, 0.68$, with $I = [0.34, 0.35]$ and $J = [0.67, 0.68]$. It can be verified that these have the required densities.

|       | $[0, 0.34]$ | $[0.34, 0.35]$ | $[0.35, 0.67]$ | $[0.67, 0.68]$ | $[0.68, 1]$ |
|-------|-------------|----------------|----------------|----------------|-------------|
| $V_1$ | 0.35        | 0.01           | 0.35           | 0.01           | 0.28        |
| $V_2$ | 0.28        | 0.01           | 0.35           | 0.01           | 0.35        |
| $V_3$ | 0.35        | 0.01           | 0.28           | 0.01           | 0.35        |

By iteratively applying Lemma 4 with every Cut query, a protocol discovers with every cut query a partial rigid system, where the intervals $I$ and $J$ always have uniform density, and their length cannot be diminished by a factor larger than 100 in each iteration. By Lemma 2, if a protocol encounters a partial rigid measure system for which there are no cuts inside $I$ and $J$, where $|I| = |J| = k$, then any configuration attainable with the existing cuts leads to envy of at least $0.01k$. To get $\epsilon$-envy, we need $k/100 < \epsilon$, and so the number of queries is $\Omega \left( \log \frac{1}{\epsilon} \right)$.

The construction can be extended to give a lower bound for any number of players.

**Theorem 4.3. Computing a connected $\epsilon$-envy-free allocation for $n \geq 3$ players requires $\Omega \left( \log \frac{1}{\epsilon} \right)$ queries.**

### 5 Perfect Allocations

Approximately perfect allocations can be computed with $O(n^2/\epsilon)$ queries. This is a consequence of Corollary 1 by taking $k = n$. For two players, the problem of computing $\epsilon$-perfect allocations can be solved more efficiently using binary search.
Theorem 5.1. An $\epsilon$-perfect allocation for two players can be computed with $O(\log \frac{1}{\epsilon})$ queries.

Proof. The main idea is to simulate Austin’s moving knife procedure in the Robertson-Webb query model, searching first by the valuation of the first player.

**Austin’s procedure:** A referee slowly moves a knife from left to right across the cake. At any point, a player can call stop. When a player called, a second knife is placed at the left edge of the cake. The player that shouted stop – say 1 – then moves both knives parallel to each other. While the two knives are moving, player 2 can call stop at any time. After 2 called stop, a randomly selected player gets the portion between player 1’s knives, while the other one gets the two outside pieces.

In the Robertson-Webb model, we start by asking both players to reveal the midpoint of the cake. If the midpoints coincide within $\epsilon$, we reached an $\epsilon$-perfect allocation. Otherwise, without loss of generality, assume the rightmost midpoint is reported by player 1 (the case of player 2 is similar); denote this cut by $z$. Then $V_1(0, z) = V_1(z, 1) = 1/2$, while $V_2(0, z) > 1/2$. Initialize $w = 1$.

![Figure 3: Approximate computation of $\epsilon$-perfect partition. Maintain two points $z$ and $w$, such that the rightmost cut must be situated in the interval $[z, w]$](image)

In the Robertson-Webb model we maintain the following invariant:

(a) There exist cut points $0 \leq z < w \leq 1$, such that the piece $[a, z]$ for which $V_1(a, z) = 1/2$ is worth strictly more than $1/2 + \epsilon$ to player 2, while the piece $[b, w]$ for which $V_1(b, w) = 1/2$ is worth strictly less than $1/2 - \epsilon$ to player 2.

Iteratively, given points $w, z$ satisfying property (a), such that $V_1(w, z) \geq \epsilon$, ask player 1 a Cut query to determine the midpoint $m$ of $[w, z]$, i.e. such that $V_1(z, m) = V_2(m, w)$, and then find through another Cut query the point $m'$ for which $V_1(m', m) = 1/2$. If there exists an $\epsilon$-perfect allocation with cuts $m$ and $m'$ then output it. Otherwise, if $V_2(m', m) > 1/2$, set $z = m$. Else, it must be the case that $V_2(m', m) < 1/2$; set $w = m$.

Each step requires a constant number of queries, and the number of iterations is $O(\log \frac{1}{\epsilon})$. 18
If the interval \([w, z]\) is worth strictly less than \(\epsilon\) to player 1, but an \(\epsilon\)-perfect allocation has not been found, let \(a\) be such that \(V_1(a, w) = 1/2\). Any partition with cuts \(a\) and \(x \in [w, z]\) is \(\epsilon\)-perfect for player 1. Then we can search for \(x \in [w, z]\) using the valuation of player 2, halving the interval \([w, z]\) in each round. A solution is guaranteed to exist and the maximum number of queries addressed to player 2 is \(O\left(\log \frac{1}{\epsilon}\right)\). □

As we show next, this bound is optimal.

**Theorem 5.2.** Computing an \(\epsilon\)-perfect allocation with the minimum number of cuts for two players requires \(\Omega\left(\log \frac{1}{\epsilon}\right)\) queries.

We prove the lower bound by maintaining throughout the execution of any protocol two intervals in which the cuts of the perfect allocation must be situated, such that the distance to a perfect partition cannot decrease too much with any cut query.

| Interval     | \([0, x]\)  | \([x, x+a]\) | \([x+a, y]\) | \([y, y+a]\) | \([y+a, 1]\) |
|--------------|-------------|--------------|--------------|--------------|--------------|
| \(V_1\)     | \(x\)      | \(a\)       | \(0.5 - a\)  | \(a\)        | \(b\)        |
| \(V_2\)     | \(c\)      | \(d\)       | \(0.5 - 2d\) | \(3d\)       | \(e\)        |

Table 6: Player 1 has uniform utility; \(y = x + 0.5, 0 < a, d \leq 0.1, x, b, c, e > 0, x + a + b = 0.5\) and \(c + 2d + e = 0.5\).

**Lemma 5.** Consider a two player problem where the valuations are consistent with Table 5 and known outside the intervals \(I = [x, x+a]\) and \(J = [y, y+a]\), such that

1. \(\epsilon < 0.001 \min\{a, b\}\).
2. any allocation obtained with cuts \(0 < k < \ell < 1\) that is \(\epsilon\)-perfect from the point of view of player 1 is worth to player 2 less than \(0.5 - d/100 - \epsilon\) when \(k < x\) and more than \(0.5 + d/100 + \epsilon\) when \(k > x + a\).

Then a new query can be handled so that the valuations remain consistent with Table 5, such that condition 2 still holds with respect to intervals \(I' = [x', x' + a'], J' = [y', y' + a']\) and parameters \(x', a' = a/100, d' = d/100\).

The proof of the lemma is included in Appendix 8.

**Proof.** (of Theorem 5.2) Let the initial configuration be defined as follows, where \(a = d = 0.1, x = 0.2\), and the valuations are constant and known everywhere outside \(I = [0.2, 0.3]\) and \(J = [0.7, 0.8]\).

Consider any partition \(A\) defined by cut points \(0 < k < \ell < 1\). Whenever \(\epsilon < 0.3\), if \(A\) is \(\epsilon\) perfect from the point of view of player 1, then the middle piece is worth
By iteratively applying Lemma 5 with every cut query received, we obtain that no \( \epsilon \)-perfect partition can be found as long as \( \epsilon < 0.001 \min\{a, b\} \), which implies the number of rounds is \( \Omega \left( \log \frac{1}{\epsilon} \right) \). \hfill \square

6 Equitable Allocations

Cechlarova, Dobos, and Pillarova [12] showed that for any number of players and any order, there exists a connected equitable allocation in that order. Moreover, the equitable allocation is proportional for some order. We give a lower bound on the number of queries required for finding connected \( \epsilon \)-equitable allocations, which matches the upper bound given by Cechlarova and Pillarova [13].

Theorem 6.1. Computing a connected \( \epsilon \)-equitable allocation for two players requires \( \Omega \left( \log \frac{1}{\epsilon} \right) \) queries.

We start with a general observation about two players, which is that the connected equitable and proportional allocation is unique for hungry valuations.

Lemma 6. For two players with hungry valuations, the cut point of the equitable allocation is unique.

Proof. By [12], there exists a unique equitable allocation for each order of the players. Let \( x \) be the cut point of the equitable allocation when the player order is \((1, 2)\). Then there exists \( c \) such that \( V_1(0, x) = V_2(x, 1) = c \), and so \( V_2(0, x) = V_1(x, 1) = 1 - c \). Thus the cut point of the equitable allocation is the same for each order of the players. \hfill \square

Next we show that when the valuations of the players are as in the next table and no cuts may be used from the interval \((x, y)\), the distance from a connected equitable allocation is high.

|          | \([0, 0.2]\) | \([0.2, 0.3]\) | \([0.3, 0.7]\) | \([0.7, 0.8]\) | \([0.8, 1]\) |
|----------|-------------|-------------|-------------|-------------|-------------|
| \( V_1 \) | 0.2         | 0.1         | 0.4         | 0.1         | 0.2         |
| \( V_2 \) | 0.15        | 0.1         | 0.3         | 0.3         | 0.15        |

Table 7: Measure system for two players where the distance from a connected equitable and proportional allocation is \( b - a \), where \( 0 < a < b < 0.5 \) and \( 0 < x < y < 1 \).
Lemma 7. Consider a two player problem where there exist points $0 < x < y < 1$ and values $0 < a < b < 0.5$ such that $V_1(0, x) = 0.5 + a = V_2(y, 1)$, $V_2(x, 1) = 0.5 + b = V_1(0, y)$. Then any connected allocation that can be formed with cut points outside the interval $(x, y)$ has distance at least $b - a$ from equitability.

Proof. Consider first allocations that can be obtained by cutting at $x$. If the player order is $(1, 2)$, then $|V_1(0, x) - V_2(x, 1)| = |(0.5 + a) - (0.5 + b)| = b - a$. Moreover, for any point $0 < z < x$, the order $(1, 2)$ gives player 1 less than $w + a$ and player 2 more than $w + b$, which leads to distance at least $b - a$ from equitability. On the other hand, if the player order is $(2, 1)$, then $V_2(0, x) = 0.5 - b$, while $V_1(x, 1) = 0.5 - a$. This allocation has distance $|V_2(0, x) - V_1(x, 1)| = b - a$ from equitability. The distance only increases for any cut $z < x$, since $|V_2(0, z) - V_1(z, 1)| = V_1(z, 1) - V_2(0, z) > (0.5 - a) - (0.5 - b) = b - a$.

Similarly, if the cut is at $y$, then the player order $(1, 2)$ gives distance $|V_1(0, y) - V_2(y, 1)| = |(0.5 + b) - (0.5 + a)| = b - a$. For any cut point $z > y$, the distance only increases, since $a < b$ and player 1 gets more than $0.5 + b$, while player 2 gets less than $0.5 + a$. Finally, if the player order is $(2, 1)$, the cut point $y$ leads to distance $|V_2(0, y) - V_1(y, 1)| = |(0.5 - a) - (0.5 - b)| = b - a$. For any cut point $z > y$, $|V_2(0, z) - V_1(z, 1)| = V_2(0, z) - V_1(z, 1) > (0.5 - a) - (0.5 - b) = b - a$. \hfill \Box

Next we show that given such a configuration, queries can be handled in a way that preserves the symmetry and the distance to equitability gets reduced by a constant factor.

Lemma 8. Consider a two player problem where there exist points $0 < x < y < 1$ and values $0 < a < b < 0.5$ such that $V_1(0, x) = 0.5 + a = V_2(y, 1)$, $V_2(x, 1) = 0.5 + b = V_1(0, y)$. Then any Cut query (addressed to either player) can be answered so that the new configuration has two new points $z, t$ such that $z, t \in (x, y)$, the valuations satisfy $V_1(0, z) = 0.5 + a' = V_2(t, 1)$, $V_2(z, 1) = 0.5 + b' = V_1(0, t)$, and $b' - a' \geq (b - a)/100$.

Proof. Suppose that player 1 receives a cut query $CUT(1; \alpha)$. If $\alpha < 0.5 + a$ or $\alpha > 0.5 + b$, then answer for both players in a way that is consistent with the history (e.g. uniform on the interval where the unique point determined by the answer of player 1 to the query falls). Otherwise, define new cuts $z, t$ such that $x < z < t < y$ and consider two subcases:

- $\alpha \in (0.5 + a, 0.5 + (a + b)/2]$. Let $a' = (a + b)/2 + (b - a)/100$ and $b' = (a + b)/2 + (b - a)/50$. Then $a < (a + b)/2 < a' < b' < b$.

- $\alpha \in (0.5 + (a + b)/2, 0.5 + b)$. Let $a' = (a + b)/2 - (b - a)/50, b' = (a + b)/2 - (b - a)/100$. Then $a < a' < b' < (a + b)/2 < b$.

In both cases, $b' - a' = (b - a)/100$. Set $V_1(0, z) = 0.5 + a'$, $V_2(0, z) = 0.5 - b'$, $V_1(0, t) = 0.5 + b'$, and $V_2(0, t) = 0.5 - a'$. In the first case, the answer to the query falls to the left of $z$ and the value can be set in any way consistent with the total value of player 2 for $[x, z]$, while in the second case the cut point falls to
the right of \( t \) and can similarly be handled arbitrarily on \( [t, y] \). Afterwards, fit the value of player 1 for the generated cut point in a way that is consistent with its valuation for \([0, z]\) and \([0, t]\).

If player 2 receives instead a cut query \( \text{Cut}(2; \alpha) \), then when \( \alpha < 0.5 - b \) or \( \alpha > 0.5 - a \), the query can be answered arbitrarily in a way that is consistent with the history. Otherwise, define cuts \( z, t \) such that \( x < z < t < y \) and consider the subcases:

1. \( \alpha \in (0.5 - b, 0.5 - (a + b)/2) \). Let \( a' = (a + b)/2 - (b - a)/50 \) and \( b' = (a + b)/2 - (b - a)/100 \). Then \( a < a' < b' < (a + b)/2 < b \) and \( 0.5 - (a + b)/2 < 0.5 - b' \).

2. \( \alpha \in (0.5 - (a + b)/2, \alpha < 0.5 - a) \). Consider values \( a' = (a + b)/2 + (b - a)/100, b' = (a + b)/2 + (b - a)/50 \). Then \( a < (a + b)/2 < a' < b' < b \) and \( 0.5 - a' < 0.5 - (a + b)/2 \).

We have \( b' - a' = (b - a)/100 \) and the valuations of players 1 and 2 for the pieces \([0, z]\) and \([0, t]\) can be defined as in case 1, when player 1 received the Cut query.

\(\square\)

**Proof.** (of Theorem 6.1) Start with cut points \( x = 0.4, y = 0.6 \), and values \( a = 0.05, b = 0.06 \), such that \( V_1(0, x) = 0.55, V_2(0, x) = 0.44, V_1(0, y) = 0.56, V_2(0, y) = 0.45 \). By Lemma 7, the distance to an equitable and proportional allocation by using cuts outside \((x, y)\) is at least \( b - a = 0.01 \). By applying Lemma 8 with every Cut query received, we get that the distance is reduced by a factor of 100 in every round. For \( \epsilon \)-equitability to hold in round \( k \), the condition \( 0.01/100^k \leq \epsilon \) must be met, and so the number of rounds is \( \Omega \left( \log \frac{1}{\epsilon} \right) \).

\(\square\)

## 7 Moving Knife Procedures

We will consider a family of protocols that seems to, on one hand, capture all types of protocols that have so far been called “moving knife” procedures and, on the other hand, be simple enough for a transparent simulation. An important ingredient of the definition is that knife positions must be continuous. To ensure that “cut queries” fall within the definition, we will only require continuity for valuation functions whose density is bounded from below; this will turn out not to hinder applications since we can take any valuation and add to it a low \( \delta \) density everywhere.

A protocol may have a constant number of “steps” where each “step” has the following form:

**Definition 3.** (A Moving Knife Step) There are a constant number \( K \) of Devices some of which have “a position on the cake” and are called Knives and others can have arbitrary real values and may be called Triggers. The devices are numbered 1 \( \ldots \) \( K \) and each device \( j \) is controlled by a single player \( i_j \). Each device has a real value that changes continuously as the time proceeds from 0 to 1. Thus each knife \( j \)'s value (i.e. his position on the cake) is given by some function \( 0 \leq x_j(t) \leq 1 \), while each trigger \( j \)'s value is given by a function \( x_j(t) \in \mathbb{R} \).
The requirement is that the value of each device \( j \) is a function of the time \( t \), of previous knives‘ locations \( x_1(t) \ldots x_{j-1}(t) \), and of \( i_j \)‘s valuation \( v_{i_j} \) as well as on any outcomes of previous steps of the protocol. The dependence function of each device on the valuation and on the outcomes of previous steps may be arbitrary, but the dependence on the time and on the values of previous devices must be continuous.

The outcome for the step is the following: for every trigger \( j \) that has a different sign at \( t = 0 \) and \( t = 1 \), i.e. where \( x_j(0) \cdot x_j(1) \leq 0 \), we get a time \( t_j \) such that \( x_j(t_j) = 0 \), as well as the values of all devices \( x_j'(t_j) \) at that time. (If the value of the trigger \( x_j \) happens to be monotone then the time \( t_j \) is unique, but in general there may be different such \( t_j \) and any one of them may be given.)

We will show that moving knife steps can be simulated efficiently in the Robertson-Webb model when the value density functions of the players are bounded and the values of the devices satisfy Lipschitz-continuity. Before proceeding, we introduce some notation.

For each device \( j \) and time \( t \), let \( i_1, \ldots, i_p \) be the indices of the devices that are knives among \( 1 \ldots j \). Let \( m \) be an upper bound such that \( w^j(t) \in [0, 1]^m \) denotes the vector of valuations of the players for the positions of the knives (i.e. the pieces \([0, x_{i_1}(t)], \ldots, [0, x_{i_p}(t)]\)), which are used when calculating the value of device \( j \). When \( m = n \times p \), the position of the \( j \)-th device depends on the valuation of each player \( i \) for every piece \([0, x_{i_p}]\) demarcated by the \( \ell \)-th knife, case in which \( w^j(t) = (V_1(0, x_{i_1}(t)), \ldots, V_n(0, x_{i_1}(t)), \ldots, V_1(0, x_{i_p}(t)), \ldots, V_n(0, x_{i_p}(t))\). However in general, some of the entries may be missing since not all the players are necessarily queried about the position of every knife.

Denote by \( c_j(t) = (t, x_1(t), \ldots, x_{j-1}(t), w^j(t)) \) the configuration consisting of the time, the values of the previous devices, and the valuations of the players for the pieces induced by the knives, based on which the position of device \( j \) is determined. Finally, let \( F_j : [0, 1]^{j+m} \to \mathbb{R} \) denote the function that determines the value of device \( j \) given the configuration at time \( t \), \( c_j(t) \), such that \( x_j(t) = F_j(c_j(t)) \).

**Definition 4 (k-Lipschitz moving knife step).** We say that a moving knife step is \( k \)-Lipschitz if the dependence function \( F_j : [0, 1]^{j+m} \to \mathbb{R} \) of each device \( j \) is \( k \)-Lipschitz for some constant \( k \). More formally, given inputs \( x, y \in [0, 1]^{j+m} \), we have \( |F_j(x) - F_j(y)| \leq k \|x - y\|_2 \).

Moving knife steps with Lipschitz dependence functions can be simulated efficiently in the Robertson-Webb model as follows.

**Theorem 7.1.** Consider a cake cutting problem where the value densities of the players are bounded from above and below by strictly positive constants \( \Delta \) and \( \delta \), respectively. Let \( \mathcal{M} \) be a \( k \)-Lipschitz moving knife step with devices \( 1 \ldots K \), such that

- the first device is a knife, whose position is bounded from below by time (i.e. for any times \( s < t \), the difference in positions is \( |x_1(s) - x_1(t)| \geq \delta |s - t| \)), and
- given the time and values of devices \( 1, \ldots, j - 1 \), the value of device \( j \) can be computed with at most \( \ell \) Robertson-Webb queries.
Let $\epsilon > 0$ and $k, \ell, K$ be constant. Then given oracle access to the inverse function $x_1^{-1} : [0, 1] \to [0, 1]$ that maps the position of the first knife to the first time when $\mathcal{M}$ reaches it, we can discover for each trigger $j$, by using $O \left( \log \frac{1}{\epsilon} \right)$ Robertson-Webb queries, a time $t_j$, such that:

- trigger $j$ is approximately zero at this time (i.e., $-\epsilon \leq x_j(t_j) \leq \epsilon$)
- there exists $\hat{t}_j \in (t_j - \epsilon, t_j + \epsilon)$ such that $x_j(\hat{t}_j) = 0$ and $|x_i(t_j) - x_i(\hat{t}_j)| < \epsilon$ for each device $i = 1 \ldots K$.

**Proof.** Let $\zeta = \max\{1, k, \Delta\}$. Recall that since each device $i$ is discovered with at most $\ell$ Robertson-Webb queries, the number of entries in $w_i(t)$ can be assumed to equal $m \leq \ell \cdot K$. We show by induction on the number of devices, that for each device $i = 1 \ldots K$ and times $s < t$, the following inequality holds:

$$|x_i(t) - x_i(s)| \leq \zeta^{2(i-1)+1} \sqrt{(m + K)^i} |t - s|. \quad (1)$$

By the condition in the lemma, for $j = 1$, we have that $|x_1(s) - x_1(t)| \leq k \cdot |t - s| \leq \zeta |t - s|$. The base case is when $j = 2$. Since the valuations of the players are bounded from above by $\zeta$, we have $V_i(x_1(s), x_1(t)) = |V_i(0, x_1(s)) - V_i(0, x_1(t))| \leq \Delta \cdot |x_1(s) - x_1(t)| \leq \zeta^2 \cdot |t - s|$. We obtain that each entry $q$ of $w^2(t)$ satisfies the inequality: $w^2_q(t) \leq \zeta^2 \cdot |t - s|$, and so

$$|F_2(c_2(t)) - F_2(c_2(s))| \leq k \cdot ||c_2(t) - c_2(s)||_2 \\
\leq k \sqrt{m + K} \cdot \max \left\{ |s - t|, |x_1(s) - x_1(t)|, \max_{q=1 \ldots m} |w^2_q(t) - w^2_q(s)| \right\} \\
\leq \zeta^3 \sqrt{m + K} |t - s|$$

Suppose the induction hypothesis holds for all devices $1 \leq i \leq j - 1$. Then

$$|x_i(t) - x_i(s)| \leq \zeta^{2(j-2)+1} \sqrt{(m + K)^{j-1}} |t - s|.$$ 

Denote by $i_1, \ldots, i_p$ the indices of the devices that are knives among $1 \ldots j - 1$. The value of device $j$ at time $t$ is $x_j(t) = F_j(t, x_1(t), \ldots, x_{j-1}(t), w^j(t))$. For each knife $1 \leq i_q \leq j - 1$, the valuation of each player $i$ satisfies

$$V_i(x_{i_q}(s), x_{i_q}(t)) \leq \zeta \cdot |x_{i_q}(s) - x_{i_q}(t)| \leq \zeta^{2j-2} \cdot \sqrt{(m + K)^{j-1}} |t - s|$$

It follows that\n
$$\max_{q=1 \ldots m} w^j_q(t) \leq \zeta^{2j-2} \cdot \sqrt{(m + K)^{j-1}} |t - s|. \text{ We get}$$

$$|F_j(c_j(t)) - F_j(c_j(s))| \leq k \cdot ||c_j(t) - c_j(s)||_2 \\
\leq k \sqrt{m + K} \cdot \max \left\{ |s - t|, |x_{i_1}(s) - x_{i_1}(t)|, \ldots, |x_{i_p}(s) - x_{i_p}(t)|, \max_{q} w^j_q(t) \right\} \\
\leq \zeta^{2(j-1)+1} \sqrt{(m + K)^j} |t - s|$$
This completes the inductive step.

Now we can simulate the procedure $\mathcal{M}$. Initialize the position of the first device (knife) when the time is 0 and 1 respectively, by setting $a = x_1(0)$ and $b = x_1(1)$. Let $j$ be any trigger of $\mathcal{M}$, where its values at time 0 and 1 are $y = x_j(0)$ and $y' = x_j(1)$, respectively.

Iteratively, ask player 1 a Cut query to identify the cut at which the interval $[a, b]$ is halved in its estimation; that is, let $w = (V_1(a) + V_1(b))/2$ and $z = \text{Cut}_1(w)$. Intuitively this corresponds to checking the configuration when half of the time has elapsed, but since there is no continuous time in the Robertson-Webb model, we use the valuation of player 1 as a proxy. Denote by $\tilde{t} = x_j^{-1}(z)$ the time when $\mathcal{M}$ sets the first knife to position $z$. By the conditions in the lemma, given the time $\tilde{t}$ and the position $z$ of the first device at $\tilde{t}$, we can iteratively find using at most $\ell$ queries the value of each device 2, $\ldots$, $K$, given the current time and the values of the previous devices. Let $x_j(\tilde{t})$ be the value attained by trigger $j$ at position $z$. If $x_j(\tilde{t}) \cdot y > 0$, update $y = z$. Else, update $y' = z$. Stop after $2 \log \left( \frac{1}{\epsilon} \right)$ steps, where $\epsilon' = \frac{\epsilon \delta^2}{2^{n+1} \sqrt{(m+K)}}$.

Since the value of player 1 for the interval $[a, b]$ halves with each iteration, we get that the final interval $[a, b]$ is very small in player 1’s estimation: $|V_1(a) - V_1(b)| \leq \epsilon'$. The value densities of the players are bounded from below by $\delta$, thus $|V_1(a) - V_1(b)| \geq \delta |a - b|$, which implies $|a - b| \leq 1/\delta \cdot |V_1(a) - V_1(b)| \leq \epsilon'/\delta$. Let $s = x_1^{-1}(a)$ and $t = x_1^{-1}(b)$ denote the times at which the first knife has positions $a$ and $b$, respectively. Then $|a - b| \geq \delta |t - s|$. By inequality 1, and using the fact that the position of the first knife is lower bounded by time, we have for each device $i$:

$$|x_i(s) - x_i(t)| \leq \zeta^{2i-1} \sqrt{(m+K)^i} |t - s|$$

$$\leq \zeta^{2i-1} \sqrt{(m+K)^i} \cdot \frac{|a - b|}{\delta}$$

$$\leq \zeta^{2i-1} \sqrt{(m+K)^i} \cdot \frac{\epsilon'}{\delta^2} < \epsilon$$

This inequality holds in particular for device $j$, case in which $|y - y'| = |x_j(s) - x_j(t)| < \epsilon$. There are two cases left. If $y > 0$ and $y' < 0$, then $0 < y \leq y' + \epsilon < \epsilon$, and we can return the time $s$. Otherwise, $y < 0$ and $y' > 0$, and so $0 < y' < y + \epsilon < \epsilon$, case in which we return time $t$. Thus for each trigger $j$ we can find an approximate solution with $O(\log \frac{1}{\epsilon})$ Robertson-Webb queries, such that the positions of the other devices are also $\epsilon$-close to their positions at a nearby time where trigger $j$ is exactly zero. This completes the proof.

The known moving knife procedures include the Dubins-Spanier procedure (which is equivalent to a discrete RW protocol), Austin’s procedure for computing perfect allocations, and several procedures for computing envy-free allocations due to Stromquist, Barbanel-Brams, Webb, Brams-Taylor-Zwicker, Levmore-Cook (the 1D version).

We also note that in fact the theorem gives a runtime that is logarithmic in $1/\epsilon$ even if the number of devices is not constant, but the valuation of the desired trigger is a function of a constant number of devices.
Corollary 2. The known moving knife procedures can be simulated with \( O \left( \log \frac{1}{\epsilon} \right) \) Robertson-Webb queries when the value density functions of the players are bounded from above by \( \Delta \) and from below by \( \delta > 0 \), where \( \delta, \Delta \) are constants.

Proof. First note that all the known procedures have a constant number of players, their moving knife steps have a constant number of devices, and the first device is a knife (held by the referee), whose inverse function is trivially known since its position is either equal to the time \( t \) or equal to \( C - t \) for some constant \( C \). The dependence functions of the known protocols are in the case of the triggers, simple linear functions (that trivially meet the Lipschitz condition) of the form \( V_i(S) - V_i(T) \) for some player \( i \) and pieces \( S, T \), and in the case of each knife, the outcomes of a Cut query, where the value given to the query is equal to the value of a player for some demarcated interval. Finally, we can reach within \( \epsilon \) the same positions of the knives as in the continuous procedure, thus approximating within \( \epsilon \) the value of each player for every piece demarcated by two adjacent knives.

We explain at a high level how to encode Austin’s procedure in the framework described in Definition 3. For Austin’s procedure we will use several moving knife steps. The first step of Austin’s procedure is discrete, and can be implemented with a Cut query in the Robertson-Webb model, in which player 1 is asked to cut the cake in half; let \( z = \text{Cut}_1(0.5) \). Then ask player 2 to evaluate the generated piece: \( \alpha = \text{Eval}_2(z) \). If \( \alpha = 0.5 \), we have found an exact perfect allocation.

Otherwise, w.l.o.g. assume \( V_2(0, z) < 0.5 \) (the other case is similar). We construct a moving knife step with three devices, where the first device will be a knife and represents the value of the referee knife. Its value (i.e. position) is equal to the time that has elapsed as long as the time is less than \( z \), and equal to \( z \) otherwise. The second device is also a knife, and its value can be obtained with an Eval and Cut query to player 1 as follows: let \( v = \text{Eval}_1(z) \) and set the value of the second knife to \( p = \text{Cut}_1(0.5 + v) \). The third device is a trigger and its value is equal to \( V_1(z, p) \), which can be obtained with a single evaluation query to player 2, \( \text{Eval}_2(z, p) \).

Note that \( K = 3 \) and \( \ell = 2 \), where \( K \) is the number of devices and \( \ell \) the number of queries required to find the value of a device given the values of the previous devices. By Lemma 7.1, we can simulate the procedure in the Robertson-Webb model to find a configuration where the trigger has value zero with \( O \left( \log \frac{1}{\epsilon} \right) \) queries, which corresponds to an \( \epsilon \)-perfect allocation.

For the Barbanel-Brams procedure, we need to keep track of six devices, such that the first device is the knife (corresponding to the referee’s knife) with position \( x_1 \), the second device is also a knife, the position \( x_2 \) of which is determined by asking player 1 a Cut query, the third device is a trigger with valuation determined by player 2’s estimation of the difference between pieces \( [0, x_1] \) and \( [x_1, x_2] \) (i.e. \( V_2(0, x_1) - V_1(x_1, x_2) \)), the fourth device is a trigger with valuation \( V_2(x_1, x_2) - V_2(x_2, 1) \), the fifth and sixth devices are triggers similar to triggers three and four, with the difference that their values are determined by player 3 instead of player 2.
Note that our simulations of the Barbanel-Brams and Austin procedures do not require that the valuations are bounded at all, since we use the specific formulations of the value functions of the triggers to eliminate the players one by one from consideration. This is not necessarily possible when the dependence functions of the devices are more complex.

**Moving Knife Step for Equitable Allocations.** Next we show a simple moving knife step in the Robertson-Webb model for computing equitable allocations for any number of hungry players, which can be implemented with \( n + 1 \) devices, one of which is a trigger and the remainder of \( n \) are knives. This moving knife step illustrates a trigger whose valuation function depends on time and the physical location of the knives (rather than the values of the players for the pieces demarcated by the knives, as is common in the previous examples).

A more complex moving knife procedure for computing exact equitable allocations that works even when the valuations are not hungry was given by Segal-Halevi and Sziklai [29]. Their protocol is based on the same idea of multiple knives moving in parallel and has more steps designed to detect regions valued at zero by some players. Note the existence of connected equitable allocations for any order of the players was established by Cechlarova, Dobos, and Pillarova [12], with a simplified proof based on the Borsuk-Ulam theorem given by Cheze [14].

**Moving Knife Step 1**: A referee slides a knife continuously across the cake, from 0 to 1. For each position \( x_1 \) of the referee’s knife, player 1 is asked for its value for the piece \([0, x_1]\); then each player \( i = 2 \ldots n \) iteratively positions its own knife at a point \( x_i \geq x_{i-1} \) with \( V_i(x_{i-1}, x_i) = V_1(0, x_1) \). Player \( n \) shouts stop when its own knife reaches the right endpoint of the cake (i.e., \( x_n = 1 \)). The cake is allocated in the order \( 1 \ldots n \), with cuts at \( x_1 \ldots x_{n-1} \).

Clearly the value of player 1 for the piece \([0, x_1]\) can be obtained with an Eval query, and the position of each knife \( j = 2 \ldots n \) can be obtained from the position of knife \( j - 1 \) by using an Eval query addressed to player \( j \) for the piece \([0, x_{j-1}]\) (i.e., \( \alpha := Eval_j(x_{j-1}) \)), followed by a Cut query: \( Cut_j(\alpha) \).

To also implement the trigger in the framework of Definition 3, its valuation function can be set as a function of time and the positions of the devices. The intuition for how to define this function can be seen by considering an augmented resource, of length \( n \), where the valuations of the players on \([1, n]\) are uniform; define the valuation of the trigger as the hypothetical position of knife \( n \) on the augmented cake (normalized to obtain a trigger value always in \([-1, 1]\)). Since the valuation functions of the devices are continuous, the \( n \)th knife must pass through the point 1 by the intermediate value theorem, which implies the trigger is zero at some time; the configuration of the knives at that time gives an equitable allocation.

**Theorem 7.2.** Moving Knife Step 1 computes a connected equitable allocation for hungry players.
8 Acknowledgements

This project has received funding from the European Research Council (ERC) under the European Unions Horizon 2020 research and innovation programme (grant agreement No 740282). The project was also supported by the ISF grant 1435/14 administered by the Israeli Academy of Sciences and Israel-USA Binational Science Foundation (BSF) grant 2014389. Simina also acknowledges support from the I-CORE Program of the Planning and Budgeting Committee and The Israel Science Foundation.

References

[1] Noga Alon. Splitting necklaces. *Advances in Mathematics*, 63(3):247 – 253, 1987.

[2] A.K. Austin. Sharing a cake. *The Mathematical Gazette*, 66(437):212215, 1982.

[3] Haris Aziz and Simon Mackenzie. A discrete and bounded envy-free cake cutting protocol for any number of agents. In *Proceedings of the 57th Symposium on Foundations of Computer Science (FOCS)*, pages 416–427, 2016.

[4] Eric Balkanski, Simina Branzei, David Kurokawa, and Ariel D. Procaccia. Simultaneous cake cutting. In *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence (AAAI)*, pages 566–572, 2014.

[5] J. Barbanel and S. Brams. Cake division with minimal cuts: Envy-free procedures for 3 persons, 4 persons, and beyond. 2004.

[6] S. Brams and A. Taylor. *Fair Division: from cake cutting to dispute resolution*. Cambridge University Press, Cambridge, 1996.

[7] Steven Brams, Alan D. Taylor, and William S. Zwicker. A moving-knife solution to the four-person envy-free cake-division problem. *Proceedings of the American Mathematical Society*, 125(2):547–554, 1997.

[8] Steven J. Brams and Alan D. Taylor. An envy-free cake division protocol. *The American Mathematical Monthly*, 102(1):9–18, 1995.

[9] Felix Brandt, Vincent Conitzer, Ulle Endriss, Jérôme Lang, and Ariel D. Procaccia, editors. *Handbook of Computational Social Choice*. Cambridge University Press, 1 edition, 2016.

[10] Simina Brânzei. A note on envy-free cake cutting with polynomial valuations. *Information Processing Letters*, 115(2):93–95, 2015.
[11] Simina Branzei and Peter Bro Miltersen. A dictatorship theorem for cake cutting. In In Proceedings of the 24th International Joint Conference on Artificial Intelligence (IJCAI), pages 482–488, 2015.

[12] Katarina Cechlarova, Jozef Dobos, and Eva Pillarova. On the existence of equitable cake divisions. Information Sciences, 228:239245, 2013.

[13] Katarina Cechlarova and Eva Pillarova. On the computability of equitable divisions. Discrete Optimization, 9:249257, 2012.

[14] Guillaume Chze. Existence of a simple and equitable fair division: A short proof. Mathematical Social Sciences, 87:92 – 93, 2017.

[15] Xiaotie Deng, Qi Qi, and Amin Saberi. Algorithmic solutions for envy-free cake cutting. Operations Research, 60(6):1461–1476, 2012.

[16] L. E. Dubins and E. H. Spanier. How to cut a cake fairly. The American Mathematical Monthly, 68(1):1–17, 1961.

[17] J. Edmonds and K. Pruhs. Balanced allocations of cake. In Proceedings of the 47th Symposium on Foundations of Computer Science (FOCS), pages 623–634, 2006.

[18] J. Edmonds and K. Pruhs. Cake cutting really is not a piece of cake. In Proceedings of the 17th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA), pages 271–278, 2006.

[19] S. Even and A. Paz. A note on cake cutting. Discrete Applied Mathematics, 7(3):285 – 296, 1984.

[20] George Gamow and Marvin Stern. Puzzle math. Viking Adult, 1958.

[21] David Kurokawa, John K. Lai, and Ariel D. Procaccia. How to cut a cake before the party ends. In Proceedings of the 27th Conference on Artificial Intelligence (AAAI), pages 555–561, 2013.

[22] H. Moulin. Fair Division and Collective Welfare. The MIT Press, 2003.

[23] Jerzy Neyman. Un theoreme dexistence. C. R. Acad. Sci. Paris, 222:843845, 1946.

[24] Ariel D. Procaccia. Thou shalt covet thy neighbors cake. In In Proceedings of the 21st International Joint Conference on Artificial Intelligence (IJCAI), page 239244, 2009.

[25] Ariel D. Procaccia. Cake cutting: Not just childs play. Communications of the ACM, 56(7):7887, 2013.

[26] Ariel D. Procaccia and Junxing Wang. A lower bound for equitable cake cutting. In Proceedings of the 18th ACM Conference on Economics and Computation (EC), 2017.

[27] J. M. Robertson and W. A. Webb. Cake Cutting Algorithms: Be Fair If You Can. A. K. Peters, 1998.
Appendix: Envy-Free Allocations

In this section we provide the remainder of the proof for Lemma 4 and the proof of Theorem 4.3.

Lemma 4 (restated) Let $A$ be a Robertson-Webb protocol and suppose that at some point during its execution, the valuations and cuts discovered are consistent with a partial rigid measure system as given in Table 2, such that the intervals $I = [x, x']$ and $J = [y, y']$ have value $k$ to all the players, uniform density, and no cut points inside.

Then if any player $i$ receives a cut query inside interval $I$, there exist answers so that the new configuration is still consistent with a partial rigid measure system, with new intervals $I'$ and $J'$ of uniform density for all the players, length $0.01k$, and with no cut points inside $I'$ and $J'$. 

30
Here we examine the scenarios where the cut query is addressed to players 2 and 3, respectively.

**Part b of Lemma 4**

**Proof.** In this case protocol $\mathcal{A}$ addresses a query to player 2. Let the query be $Cut(2; \alpha)$. We have two cases:

**Case 1:** $\alpha \in (m_2, m_2 + 2k/3]$. We will hide the interval at the right of the point $x + 2k/3$. Define $m = x + 2k/3$ and $n = x + 2k/3 + 0.01k$. Set $I' = [m, n]$. Let $p = y + k/3$ and $q = y + k/3 + 0.01k$. Set $J' = [p, q]$. Let the values of all the players be $0.01k$ in the intervals $I'$ and $J'$. Since the length of these intervals is exactly $0.01k$, it follows that everyone’s densities are uniform herein.

Set the density of player 2 uniform on each of the generated subintervals. Then $0.97k/3 + \ell_2 + k/3 = 1.97k/3 + \ell_2$. Thus for player 2 the values are consistent with a partial rigid measure system. Set $\ell'_2 = \ell_2 + 1.97k/3$ and $m'_2 = m_2 + 2k/3$.

We must now fit the valuations of players 1 and 3, which implies again finding parameters $0 < \lambda, \mu, \phi, \psi < 0.99$ so that the matrix of valuations is as follows.

|     | $[0, x]$ | $[x, m]$ | $[m, n]$ | $[n, x + k]$ | $[x + k, y]$ | $[y, p]$ | $[p, q]$ | $[q, y + k]$ | $[y + k, 1]$ |
|-----|----------|----------|----------|-------------|-------------|----------|----------|-------------|-------------|
| $V_2$ | $m_2$    | $2k/3$   | $0.01k$  | $0.97k/3$   | $\ell_2$   | $k/3$    | $0.01k$  | $1.97k/3$   | $\ell_2$    |
| $V_1$ | $\ell_1$ | $0.99k - \lambda k$ | $0.01k$ | $\lambda k$ | $\ell_1$   | $\mu k$  | $0.01k$  | $0.99k - \mu k$ | $m_1$       |
| $V_3$ | $\ell_3$ | $0.99k - \phi k$ | $0.01k$ | $\phi k$   | $m_3$      | $\psi k$ | $0.01k$  | $0.99k - \psi k$ | $\ell_3$    |

For the valuation of player 1 to be consistent with a partial rigid measure system, $\ell_1 + 0.99k - \lambda k = \lambda k + \ell_1 + \mu k \iff \mu = 0.99 - 2\lambda$. The densities of player 1 on the new intervals are:

- $[x, m]$: $V_1(x, m)/(m - x) = (0.99k - \lambda k)/(2k/3) = 297/200 - 3\lambda/2$
- $[n, x + k]$: $V_1(n, x + k)/(x + k - n) = \lambda k/(0.97k/3) = 300\lambda/97$
- $[y, p]$: $V_1(y, p)/(p - y) = \mu k/(k/3) = 3\mu = 297/100 - 6\lambda$
- $[q, y + k]$: $V_1(q, y + k)/(y + k - q) = (0.99k - \mu k)/(1.97k/3) = 600\lambda/197$

To ensure that all these densities are in the interval $(1/\sqrt{2}, \sqrt{2})$, we can set $\lambda = 0.26$. Then $\mu = 0.47$. Update $\ell'_1 = \ell_1 + 0.73k$ and $m'_1 = m_1 + 0.52k$.

For player 3 we have $\phi = \psi$. The densities of player 3 are:

- $[x, m]$: $V_3(x, m)/(m - x) = (0.99k - \phi k)/(2k/3) = 297/200 - 3\phi/2$
- $[n, x + k]$: $V_3(n, x + k)/(x + k - n) = \phi k/(0.97k/3) = 300\phi/97$
• \( [y,p] : V_3(y,p)/(p-y) = \phi k/(k/3) = 3\phi \)

• \( [q,y+k] : V_3(q,y+k)/(y+k-q) = (0.99k - \phi k)/(1.97k/3) \)

Setting \( \phi = \psi = 0.25 \) ensures all the values are in \((1/\sqrt{2}, \sqrt{2})\). Then \( \ell_3' = \ell_3 + 0.75k \) and \( m'_3 = m_3 + 0.5k. \)

**Case 2: \( \alpha \in (m_2 + 2k/3, m_2 + k) \).** This time we will hide the interval to the left of the point \( x + 2k/3 \).

Define \( m = x + 2k/3 - 0.01k, n = x + 2k/3, p = y + 1.97k/6, \) and \( q = y + 2.03k/6. \) Set \( I' = [m,n] \) and \( J' = [p,q] \) and let the densities of all the players be uniform on \( I' \) and \( J' \). It can be verified that \( k/3 + \ell_2 + 1.97k/6 = 3.97k/6 + \ell_2 \); moreover its density is uniform on all the new intervals. Update \( m'_2 = m_2 + 2k/3 - 0.01k \) and \( \ell'_2 = \ell_2 + 3.97k/6 \).

The goal is to fit the valuations of players 1 and 3, which implies computing values \( 0 < \lambda, \mu, \phi, \psi < 0.99 \) so that the matrix of valuations is as follows.

|       | \([0,x]\) | \([x,m]\) | \([m,n]\) | \([n,x+k]\) | \([x+k,y]\) | \([y,p]\) | \([p,q]\) | \([q,y+k]\) | \([y+k,1]\) |
|-------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|
| \(V_2\) \(m_2\) | 2k/3 - 0.01k | 0.01k | \(k/3\) | \(\ell_2\) | 1.97k/6 | 0.01k | 3.97k/6 | \(\ell_2\) |       |
| \(V_1\) \(\ell_1\) | 0.99k - \(\lambda\)k | 0.01k | \(\lambda\)k | \(\ell_1\) | \(\mu\)k | 0.01k | 0.99k - \(\mu\)k | \(m_1\) |       |
| \(V_3\) \(\ell_3\) | 0.99k - \(\phi\)k | 0.01k | \(\phi\)k | \(m_3\) | \(\psi\)k | 0.01k | 0.99k - \(\psi\)k | \(\ell_3\) |       |

For player 1 we have \( \mu = 0.99 - 2\lambda \) and the next densities:

• \( [x,m] : V_1(x,m)/(m-x) = (0.99 - \lambda)/(197/300) \)

• \( [n,x+k] : V_1(n,x+k)/(x+k-n) = \lambda k/(k/3) = 3\lambda \)

• \( [y,p] : V_1(y,p)/(p-y) = \mu k/(1.97k/6) = 600/197 \cdot (0.99 - 2\lambda) \)

• \( [q,y+k] : V_1(q,y+k)/(y+k-q) = (0.99k - \mu k)/(3.97k/6) = 1200\lambda/297 \)

Setting \( \lambda = 0.265 \) ensures that all of these densities are in \((1/\sqrt{2}, \sqrt{2})\). Then \( \mu = 0.46. \) Update \( \ell'_1 = \ell_1 + 0.725k \) and \( m'_1 = m_1 + 0.53k. \)

For player 3 we have \( \phi = \psi \) and the following densities:

• \( [x,m] : V_3(x,m)/(m-x) = (0.99 - \phi)/(1.97/3) \)

• \( [n,x+k] : V_3(n,x+k)/(x+k-n) = \phi k/(k/3) = 3\phi \)

• \( [y,p] : V_3(y,p)/(p-y) = \phi k/(1.97k/6) = 600\phi/197 \)
\[ [q, y + k] : V_3(q, y + k)/(y + k - q) = (0.99 - \phi)/(397/600) \]

Setting \( \phi = \psi = 0.25 \) ensures all the values are in the range \( (1/\sqrt{2}, \sqrt{2}) \). Update \( \ell'_3 = \ell_3 + 0.74k \) and \( m'_3 = m_3 + 0.5k \).

Similarly to part I of the proof, the query asked by the protocol falls outside the new intervals \( I' \) and \( J' \), and so it can be answered uniformly for player 2 and proportionally to the weight on the respective interval for players 1 and 3. This completes the second part of the proof.

\[ \square \]

**Part c of Lemma 4**

**Proof.** We analyze the situation where the protocol \( \mathcal{A} \) addresses a query to player 3. Let the query be \( \text{Cut}(3; \alpha) \) and consider two cases:

**Case 1:** \( \alpha \in (\ell_3, \ell_3 + 2k/3) \). We will hide the interval at the right of the point \( x + 2k/3 \). Define \( m = x + 2k/3, n = x + 2k/3 + 0.01k, p = y + k/3 - 0.01k, \) and \( q = y + k/3 \).

Set \( I' = [m, n] \) and \( J' = [p, q] \). Let the value of each player be \( 0.01k \) for the intervals \( I' \) and \( J' \). Again all the densities are uniform on \( I' \) and \( J' \). Set the density of player 3 uniform on all the new intervals and update \( \ell'_3 = \ell_3 + 2k/3 \) and \( m'_3 = m_3 + 1.94k/3 \). The goal is to find the appropriate values for players 1 and 2, or equivalently, \( 0 < \lambda, \mu, \phi, \psi < 0.99 \). These are captured in the next table.

|       | \( [0, x] \) | \( [x, m] \) | \( [m, n] \) | \( [n, x + k] \) | \( [x + k, y] \) | \( [y, p] \) | \( [p, q] \) | \( [q, y + k] \) | \( [y + k, 1] \) |
|-------|-------------|-------------|-------------|----------------|----------------|-------------|-------------|----------------|----------------|
| \( V_3 \) | \( \ell_3 \) | \( 2k/3 \) | \( 0.01k \) | \( k/3 - 0.01k \) | \( m_3 \) | \( k/3 - 0.01k \) | \( 0.01k \) | \( 2k/3 \) | \( \ell_3 \) |
| \( V_1 \) | \( \ell_1 \) | \( 0.99k - \lambda k \) | \( 0.01k \) | \( \lambda k \) | \( \ell_1 \) | \( \mu k \) | \( 0.01k \) | \( 0.99k - \mu k \) | \( m_1 \) |
| \( V_2 \) | \( m_2 \) | \( 0.99k - \phi k \) | \( 0.01k \) | \( \phi k \) | \( \ell_2 \) | \( \psi k \) | \( 0.01k \) | \( 0.99k - \psi k \) | \( \ell_2 \) |

For player 1 we have \( \ell_1 + 0.99k - \lambda k = \lambda k + \ell_1 + \mu k \iff \mu = 0.99 - 2\lambda \). Its density is:

- \[ [x, m] : V_1(x, m)/(m - x) = (0.99k - \lambda k)/(2k/3) = 297/200 - 3\lambda/2 \]
- \[ [n, x + k] : V_1(n, x + k)/(x + k - n) = \lambda k/(k/3 - 0.01k) = 300\lambda/97 \]
- \[ [y, p] : V_1(y, p)/(p - y) = \mu k/(0.97k/3) = 300\mu/97 = 300/97 \cdot (0.99 - 2\lambda) \]
- \[ [q, y + k] : V_1(q, y + k)/(y + k - q) = (0.99k - \mu k)/(2k/3) = 3\lambda \]

Setting \( \lambda = 0.267 \) ensures the density is in \( (1/\sqrt{2}, \sqrt{2}) \) everywhere. Then \( \mu = 0.456 \). Update \( \ell'_1 = \ell_1 + 0.723k \) and \( m'_1 = m_1 + 0.534k \).

For player 2 we have \( \phi k + \ell_2 + \psi k = 0.99k - \psi k + \ell_2 \iff \psi = 99/200 - \phi/2 \).

The density of player 2 is:
Set $\phi = 0.25$. Then $\psi = 0.37$. Update $\ell_2' = \ell_2 + 0.62k$ and $m_2' = m_2 + 0.74k$.

**Case 2:** $\alpha \in (\ell_3 + 2k/3, \ell_3 + k)$. We will hide the interval at the left of the point $x + \frac{2k}{3}$. Define $m = x + 2k/3 - 0.01k$, $n = x + 2k/3$, $p = y + k/3$, and $q = y + k/3 + 0.01k$.

Set $I' = [m, n], J' = [p, q]$ Let the values of all the players be $0.01k$ for the entire intervals $I'$ and $J'$. Again all densities are uniform on $I'$ and $J'$. Set the density of player 3 uniform on all the new intervals and update $\ell_2' = \ell_3 + 2k/3 - 0.01k$ and $m_2' = m_3 + 2k/6$. The goal is to find the appropriate values for players 1 and 2, or equivalently, $0 < \lambda, \mu, \phi, \psi < 0.99$, which are captured in the next table.

|        | $[0, x]$ | $[x, m]$ | $[m, n]$ | $[n, x + k]$ | $[x + k, y]$ | $[y, p]$ | $[p, q]$ | $[q, y + k]$ | $[y + k, 1]$ |
|--------|---------|---------|---------|-------------|-------------|---------|---------|-------------|-------------|
| $V_3$  | $\ell_3$ | $2k/3 - 0.01k$ | $0.01k$ | $k/3$ | $m_3$ | $k/3$ | $0.01k$ | $2k/3 - 0.01k$ | $\ell_3$ |
| $V_1$  | $\ell_1$ | $0.99k - \lambda k$ | $0.01k$ | $\lambda k$ | $\ell_1$ | $\mu k$ | $0.01k$ | $0.99k - \mu k$ | $m_1$ |
| $V_2$  | $m_2$ | $0.99k - \phi k$ | $0.01k$ | $\phi k$ | $\ell_2$ | $\psi k$ | $0.01k$ | $0.99k - \psi k$ | $\ell_2$ |

For player 1 we get $\mu = 0.99 - 2\lambda$ and the following densities:

- $[x, m]: V_1(x, m)/(m - x) = (0.99k - \lambda k)/(2k/3 - 0.01k) = (297 - 300\lambda)/197$
- $[n, x + k]: V_1(n, x + k)/(x + k - n) = \lambda k/(k/3) = 3\lambda$
- $[y, p]: V_1(y, p)/(p - y) = \mu k/(k/3) = 3\mu = 297/100 - 6\lambda$
- $[q, y + k]: V_1(q, y + k)/(y + k - q) = (0.99k - \mu k)/(2k/3 - 0.01k) = 600\lambda/197$

Setting $\lambda = 0.26$ ensures the densities are in the required range. Then $\mu = 0.47$. Update $\ell_1' = \ell_1 + 0.73k$ and $m_1' = m_1 + 0.52k$.

Finally, we must fit the answers of player 2. We get $\psi = 99/200 - \phi/2$ and the following densities:

- $[x, m]: V_2(x, m)/(m - x) = (0.99k - \phi k)/(2k/3 - 0.01k) = (297 - 300\phi)/197$
- $[n, x + k]: V_2(n, x + k)/(x + k - n) = \phi k/(k/3) = 3\phi$
- $[y, p]: V_2(y, p)/(p - y) = \psi k/(k/3) = 3\psi = 297/200 - 3\phi/2$
• \([q, y + k] : V_2(q, y + k)/(y + k - q) = (0.99k - \psi k)/(2k/3 - 0.01k) = (99/200 + \phi/2)/(197/300)\)

Let \(\phi = 0.25\). Then \(\psi = 0.37\). Update \(\ell_2 = \ell_2 + 0.62k\) and \(m_2 = m_2 + 0.74k\).

In both cases the query to player 3 falls outside the interval \(I'\), so the query can be answered for all the players in a way that is proportional to their density on the respective interval. \(\square\)

**Theorem 4.3** (restated) Computing a connected \(\epsilon\)-envy-free allocation for \(n \geq 3\) players requires \(\Omega \left( \log \frac{1}{\epsilon} \right)\) queries.

**Proof.** For ease of exposition, we assume the number of players is divisible by 3. Let \(K = n/3\) and divide the players in disjoint sets of three, such that each group \(S_i = \{3i - 2, 3i - 1, 3i\}\), for \(i \in \{1, \ldots, K\}\), the players in \(S_i\) are only interested in the piece \(J_i = [(i - 1)/K, i/K]\), and their valuations form a generalized rigid measure system on \(J_i\) with higher densities, such that \(K/\sqrt{2} < v_j(x) < K\sqrt{2}\), for each player \(j \in S_i\) and \(x \in J_i\). By applying Lemma 2 for \(a = \sqrt{2}/K\) and \(b = K\sqrt{2}\), we get that for any two disjoint pieces \(S_1, S_2 \subset J_i\), if the valuation of player \(i\) satisfies \(V_i(S_1) \geq 2V_i(S_2)\), then the valuation of another player \(j\) in the same group as \(i\) satisfies \(2V_j(S_1) \geq V_j(S_2)\). Thus Lemma 3 still applies for each group \(S_i\) and interval \(J_i\). The queries are handled as follows. Whenever a player \(j \in S_i\) receives a cut query outside the piece they are interested in, the answer is given so as to not introduce new cut points. On the other hand, if player \(j\) receives a cut query in the interval \(J_i\), the answer is given as in the construction of Theorem 4.2, where the points are scaled to reside in \(J_i\).

Consider the final allocation computed by a Robertson-Webb algorithm \(\mathcal{A}\), and let \(x_i\) be the cut point that separates group \(S_i\) from group \(S_{i+1}\). If \(x_1 \geq 1/K\), then the allocation \(\mathcal{A}\) is \(\epsilon\)-envy-free among the players in \(S_1\) if and only if the algorithm discovers the generalized rigid measure system on \([0, 1/K]\), since the piece \([1/K, x_1]\) is worth zero to all the players in \(S_1\). Otherwise, \(x_1 \leq 1/K\). If \(x_2 \geq 2/K\), then for the allocation to be \(\epsilon\)-envy-free among the players in \(S_2\), \(\mathcal{A}\) must discover (within \(\epsilon\) error) the measure system among the players in \(S_2\) on interval \(J_2\). Otherwise, we have \(x_2 \leq 2/K\). Iteratively, we either find an interval \(i \leq K - 1\) where the algorithm must solve the problem where the solution is unique among the group \(S_i\), or reach \(i = K\) with \(x_{K-1} \leq (K - 1)/K\), case in which \(\mathcal{A}\) must find the measure system among the players \(S_K\) on \(J_K\). Since finding an \(\epsilon\)-envy-free allocation on \(J_i\) among \(S_i\) requires \(\Omega \left( \log \frac{1}{\epsilon} \right)\) queries for all \(i \in \{1, \ldots, K\}\), which implies the required lower bound. The cases where the number of players is of the form \(n = 3K + 1\) and \(n = 3K + 2\) can be solved by extending the lemmas for three players to four and five players, respectively, by observing that the cases that appear in both Lemma 3 and Lemma 5 rely on a number of combinations that are independent of the number of players (in the case of Lemma 3, whether a player gets allocated a piece with two columns, one, or none, while in the case of Lemma 5, whether the cut falls in an interval worth \(m_i\) or \(\ell_i\) to a player, and whether the new interval maintained is hidden on the left or right side of the cut). Then when \(n = 3K + 1\), the group \(S_1 = \{1, 2, 3, 4\}\), while when \(n = 3K + 2\), \(S_1 = \{1, 2, 3, 4, 5\}\). \(\square\)
Appendix: Perfect Allocations

In this section we prove the lemma that is used iteratively to hide the intervals containing the cuts of the perfect allocation.

**Lemma 5** (restated) Consider a two player problem where the valuations are consistent with Table 5 and known outside the intervals $I = [x, x + a]$ and $J = [y, y + a]$, such that

1. $\epsilon < 0.001 \min\{a, b\}$.

2. any allocation obtained with cuts $0 < k < \ell < 1$ that is $\epsilon$-perfect from the point of view of player 1 is worth to player 2 less than $0.5 - d'/100 - \epsilon$ when $k < x$ and more than $0.5 + d'/100 + \epsilon$ when $k > x + a$.

Then a new query can be handled so that the valuations remain consistent with Table 5, such that condition 2 still holds with respect to intervals $I' = [x', x' + a']$, $J' = [y', y' + a']$ and parameters $x', a' = a/100$, and $d' = d/100$.

**Proof.** The valuations are published outside $I, J$, so queries that fall outside these intervals are handled by the conditions in the lemma. Otherwise, suppose player 2 receives a query $Cut(2; \alpha)$ in one of $I, J$. The scenario where player 1 receives the query will follow from the analysis for player 2. The new intervals maintained will be $I' = [m, n]$, $J' = [p, q]$, where $m, n, p, q$ are defined depending on the query. Set $a' = a/100$, $d' = d/8$, and let $0 < k < \ell < 1$ be the defining cuts of a partition that is $\epsilon$-perfect from the point of view of player 1. Then $0.5 - \epsilon \leq \ell - k \leq 0.5 + \epsilon$. We show that if $k, \ell \notin I', J'$, then $[k, \ell]$ is either too small or too large for player 2. Consider the first scenario, where player 2’s answer is in the interval $I$.

**Case 1.a.** $\alpha \in (c, c + d/2)$. Let $m = x + a/2$, $n = x + 51a/100$, $p = y + a/2$, $q = y + 51a/100$. Publish the valuations as constant outside $I'$, $J'$ and consistent with the next table.

|       | $[0, x]$ | $[x, m]$ | $[m, n]$ | $[n, x + a]$ | $[x + a, y]$ | $[y, p]$ | $[p, q]$ | $[q, y + a]$ | $[y + a, 1]$ |
|-------|--------|--------|--------|------------|---------------|--------|--------|------------|------------|
| $V_1$ | $x$    | $a/2$  | $a/100$| $49a/100$ | $0.5 - a$     | $a/2$  | $a/100$| $49a/100$  | $b$        |
| $V_2$ | $c$    | $d/2$  | $d/8$  | $3d/8$    | $0.5 - 2d$   | $11d/8$| $3d/8$| $10d/8$    | $e$        |

Next we show that if $k \leq m$, then the piece $[k, \ell]$ is worth less than $0.5 - d'/100 - \epsilon$ to player 2, while if $k > n$, then $[k, \ell]$ is worth more than $0.5 + d'/100 + \epsilon$ to player 2. If $k < x - \epsilon$, then $\ell \leq y$, and the claim holds by the assumption in the lemma’s statement. If $k > x + \epsilon$, then $\ell \geq y + a$, and the lemma holds. Otherwise, we have a few cases:

- $k \in [x - \epsilon, x]$. Then $\ell \leq y + \epsilon$. Let $w = V_2(x - \epsilon, x)$. Since $V_2(x - \epsilon, y) = w + V_2(x, y) = \ldots$
Case 1.b. \( \alpha \in (c + d/2, c + d) \). Let \( m = x + 49a/100, n = x + a/2, p = y + 49a/100, q = y + a/2 \). Let the valuations be constant and known outside \( I', J' \) as follows.

\[
\begin{array}{|c|c|c|c|c|c|c|c|c|}
\hline
& [0, x] & [x, m] & [m, n] & [n, x + a] & [x + a, y] & [y, p] & [p, q] & [q, y + a] & [y + a, 1] \\
\hline
V_1 & x & 49a/100 & a/100 & a/2 & 0.5 - a & 49a/100 & a/100 & a/2 & b \\
V_2 & c & 3d/8 & d/8 & d/2 & 0.5 - 2d & 10d/8 & 3d/8 & 11d/8 & e \\
\hline
\end{array}
\]

We show the required discrepancy holds for the piece \([k, \ell] \). If \( k < x - \epsilon \) or \( k > x + a + \epsilon \), the claim follows by the lemma’s condition. The remaining cases are:

- \( k \in [x - \epsilon, x] \). We have \( w = V_2(x - \epsilon, x) < 99d/100 - \epsilon \). Since \( \epsilon < 0.001a \), we have \( V_2(k, \ell) \leq w + V_2(x, y) + \epsilon \cdot (10d/8)/(49a/100) < 0.5 - d'/100 - \epsilon \).
• $k \in [x, m]$. Let $\delta = k - x$. Note $V_2(m, y) = 0.5 - 11d/8$. Since $\delta \leq 49a/100$ and $\epsilon < 0.01 \min\{a, d\}$, we get

$$V_2(k, \ell) \leq (49a/100 - \delta) + 3d/8 + 0.5 - 11d/8 + (\delta + \epsilon) \cdot 10d/8 < 0.5 - d'/100 - \epsilon$$

• $k \in [n, x + a]$. Let $\delta = k - n$. We have $\ell \geq k + 0.5 - \epsilon$ and $\epsilon < 0.01 \min\{a, d\}$. If $\delta \leq \epsilon$, then

$$V_2(k, \ell) \geq V_2(n, q) - \epsilon \cdot d/a = 0.5 + d/8 - \epsilon \cdot d/a > 0.5 + d'/100 + \epsilon.$$

Otherwise, $\delta > \epsilon$, thus $\ell \in [q, y + a]$. Since $V_2(x + a, q) = 0.5 - 3d/8$, we get

$$V_2(k, \ell) \geq d/2 - \delta \cdot d/a + 0.5 - 3d/8 + (\delta - \epsilon) \cdot (11d/8)/(a/2) > 0.5 + d'/100 + \epsilon.$$

• $k \in [x + a, x + a + \epsilon]$. Using that $w = V_2(x + a, x + a + \epsilon) < 99d/100 - \epsilon$ and $\epsilon < 0.01a$, we get

$$V_2(k, \ell) \geq 0.5 + d - w - \epsilon \cdot (11d/8)/(a/2) > 0.5 + d'/100 + \epsilon.$$

The second scenario, where the answer of player 2 falls in the interval $J$, has two subcases:

**Case 2.a.** $\alpha \in (0.5 + c - d, 0.5 + c + d/2]$. Let $m = x + a/2$, $n = x + 51a/100$, $p = y + a/2$, $q = y + 51a/100$. Let the valuations be known and of constant density on the new intervals except $I', J'$.

|        | $[0, x)$ | $[x, m]$ | $[m, n]$ | $[n, x + a]$ | $[x + a, y]$ | $[y, p]$ | $[p, q]$ | $[q, y + a]$ | $[y + a, 1)$ |
|--------|----------|----------|----------|-------------|-------------|---------|---------|-------------|-------------|
| $V_1$  | $x$      | $a/2$    | $a/100$  | $49a/100$   | $0.5 - a$   | $a/100$ | $49a/100$ | $b$         |
| $V_2$  | $c$      | $5d/8$   | $d/8$    | $d/4$       | $0.5 - 2d$  | $3d/2$  | $3d/8$  | $9d/8$      | $e$         |

If $k < x - \epsilon$ or $k > x + a + \epsilon$, the discrepancy between the valuations of the players for $[k, \ell]$ holds by the lemma’s statement. If $k \in [x - \epsilon, x]$, note the change in the density of player 2 on the interval $[y, p]$ compared to Case 1.a is constant, thus a similar argument works when $\epsilon < 0.001a$. If $k \in [x + a, x + a + \epsilon]$, the claim also follows from Case 1.a, where the interval $[q, y + a]$ had the same length and higher value density for player 2 than here. The remaining cases are:

• $k \in [x, m]$. Let $\delta = k - x/2$. Then $V_2(k, \ell) \leq 0.5 - d + \delta \cdot 7d/(4a) \leq 0.5 - d/8 < 0.5 - d'/100 - \epsilon$.

• $k \in [n, x + a]$. Let $\delta = k - n$. If $\delta \leq \epsilon$, the claim follows as in Case 1.a. If $\delta > \epsilon$, then

$$V_2(k, \ell) \geq 0.5 + d/8 - \epsilon \cdot (9d/8)/(49a/100) > 0.5 + d'/100 + \epsilon.$$

**Case 2.b.** $\alpha \in (0.5 + c + d/2, 0.5 + c + 2d)$. Let $m = x + 49a/100$, $n = x + a/2$, $p = y + 49a/100$, $q = y + a/2$. Let the valuations be as follows.

If $k \leq x$, $k \in [x + a, x + a + \epsilon]$, or $k > x + a + \epsilon$, the claim follows as in the previous cases. If

• $k \in [x, m]$. Let $\delta = k - x \leq 49a/100$. Then $V_2(k, \ell) \leq 0.5 - d - \delta \cdot (d/4)/(49a/100) + (\delta + \epsilon) \cdot (9d/8)/(49a/100) \leq 0.5 - d/8 + \epsilon \cdot (9d/8)/(49a/100) \leq 0.5 - d'/100 - \epsilon$. 

38
\begin{tabular}{|c|c|c|c|c|c|c|c|c|c|c|}
\hline
 & [0, x] & [x, m] & [m, n] & [n, x + a] & [x + a, y] & [y, p] & [p, q] & [q, y + a] & [y + a, 1] \\
\hline
$V_1$ & x & 49a/100 & a/100 & a/2 & 0.5 - a & 49a/100 & a/100 & a/2 & b \\
$V_2$ & c & d/4 & d/8 & 5d/8 & 0.5 - 2d & 9d/8 & 3d/8 & 3d/2 & e \\
\hline
\end{tabular}

\begin{itemize}
\item k \in [n, x + a]. Let \( \delta = k - n \). If \( \delta \leq \epsilon \) the claim is as in Case 1.b. If \( \delta > \epsilon \),
\( V_2(k, \ell) \geq 0.5 + \frac{d}{8} + \delta \cdot \frac{290d}{(8a)} - \epsilon \cdot \frac{300d}{(8a)} > 0.5 + \frac{d'}{100} + \epsilon. \)
\end{itemize}

Thus each cut query can be answered so that the new partition is still \( \frac{d'}{100} \) far from perfect whenever player 1 believes the middle piece is almost perfect, where the values of \( a' \) and \( d' \) has been reduced by a constant factor. This completes the proof of the lemma. \qed