Computing Socially-Efficient Cake Divisions

Yonatan Aumann, Yair Dombb and Avinatan Hassidim

Abstract

We consider a setting in which a single divisible good ("cake") needs to be divided between n players, each with a possibly different valuation function over pieces of the cake. For this setting, we address the problem of finding divisions that maximize the *social welfare*, focusing on divisions where each player needs to get one contiguous piece of the cake. We show that for both the utilitarian and the egalitarian social welfare functions it is NP-hard to find the optimal division. For the utilitarian welfare, we provide a constant factor approximation algorithm, and prove that no FPTAS is possible unless P=NP. For egalitarian welfare, we prove that it is NP-hard to approximate the optimum to any factor smaller than 2. For the case where the number of players is small, we provide an FPT (fixed parameter tractable) FPTAS for both the utilitarian and the egalitarian welfare objectives.

1 Introduction

Consider a town with a central conference hall, erected by the municipality for the benefit of the townspeople. Different people and organizations wish to use the hall for their events, each for a possibly different duration. Furthermore, each such event may have its preferences and constraints on the times when it can take place, e.g. only in the evenings, on weekends, prior to some date, etc. How should the municipality allocate the hall to the different events? How do we compute the allocation that maximizes the social welfare provided by this common resource?

A natural setting for analyzing the above problem is that of *cake cutting*, where a single divisible good needs to be divided between several players with possibly different preferences regarding the different parts of the good, or "cake". The cake cutting problem was first introduced in the 1940's by Steinhaus [Ste49], where the goal was to give each of the n players "their due part", i.e. a piece worth at least $\frac{1}{n}$ of the entire cake by their own measure. (In the cake cutting literature, this fairness requirement is termed *proportionality*.) Since then, other objectives have also been considered, with the majority of them requiring that the division be "fair", under some definition of fairness (e.g. envy-freeness).

Here, we address the fundamental problem of *maximizing social welfare* in cake cutting. Given a shared resource, the valuation functions of the players for this resource, and a social welfare function, the problem is to find an allocation that maximizes the welfare. Maximizing social welfare has been previously considered for dividing a set of discrete indivisible items, each of which must be given in whole to one player. Here, we consider the problem with a *single, continuously divisible good*, and furthermore focus on the case where each player needs to get a *single contiguous piece* of the good. The contiguity requirement is natural in many settings, e.g. dividing time (as in the example above), spectrum, and real-estate.

Results. We show that the problems of maximizing utilitarian and egalitarian welfare are both NP-hard in the strong sense. For egalitarian welfare, we further show that it is hard to approximate the optimum to any factor smaller than 2.

For utilitarian welfare, we provide a constant-factor approximation algorithm (note that the strong NP-hardness result implies that no FPTAS exists for the problem). Specifically, our algorithm finds a division with utilitarian welfare within $8 + o(1)$ of the optimum,

in polynomial time. We also show that approximating both the utilitarian and egalitarian welfare is fixed-parameter-tractable with regards to the parameter n (the number of players).

Finally, we consider the case where the contiguity requirement is dropped, i.e. each player may get a *collection* of intervals. For this setting, we show that the situation varies greatly depending on the model of input. When the valuations are given explicitly to the algorithm, and are piecewise constant, the problem can be solved in polynomial time. However, if the algorithm has only oracle access to the valuations, then it is impossible to do any better than an n-factor approximation, even if the valuations themselves are piecewise uniform.

Due to space constraints, many of the proofs are deferred to the full version of the paper.

Related Work. The problem of maximizing egalitarian welfare when allocating a set of indivisible goods has been extensively considered in the last 15 years [Woe97, AAWY98, BS06, CCK09]. The currently known best algorithms are a polynomial-time algorithm achieving an approximation factor of $O(\sqrt{n}\log^3 n)$ [AS07], and an algorithm obtaining $\tilde{O}(n^\epsilon)$ approximation in time $n^{O(1/\epsilon)}$, for any $\epsilon = \Omega(\frac{\log\log n}{\log n})$ [CCK09]. Hardness of approximation for this problem, however, is proven only for a factor of 2 or less [BD05]. Better approximation guarantees are known for more restricted settings, e.g. when valuations are restricted to having only one possible non-zero value for each item [BS06, Fei08]. Envy minimization in this setting has also been considered in [LMMS04], which showed hardness results as well as an FPTAS for the case of players with identical preferences. Unlike this body of work, which considers a *non-ordered set* of indivisible items, here we consider a single divisible item, and furthermore require that each player obtain a single contiguous piece of this good.

Cake cutting problems were first introduced in the 1940's [Ste49], and were studied in many variants since then. Various algorithms were proposed for the problem, including a number of "moving knife" algorithms, which perform an infinite number of valuations by continuously moving a knife over the cake (for some examples, see [Str80, EP84] and [BT95]). In addition to the algorithmic results, there has also been work on existence theorems [DS61, Str80], lower bounds for the complexity of such algorithms ([SW03, Str08, Pro09], to mention just a few), and a number of books on the subject, e.g. [BT96, RW98].

The issue of social welfare in cake cutting was first considered in Caragiannis et al. [CKKK09] which aimed to quantify the degradation in social welfare that may be caused by different fairness requirements; the same question was studied for connected pieces in [AD10]. Guo and Conitzer [GC10], and Han et al. [HSTZ11] study the utilitarian welfare achievable by truthful mechanisms for dividing a set of divisible goods, a setting very similar to a cake with piecewise-constant valuations and non-connected pieces. Cohler et al. [CLPP11] study utilitarian welfare maximization under the envy-freeness requirement (with non-connected pieces). Bei et al. [BCH$^+$12] consider a similar question, but with connected pieces, and with proportionality replacing envy-freeness. Also related is the work of Zivan [Ziv11] which suggests a way for increasing utilitarian welfare using trust.

2   Model and Definitions

Valuation Functions. In our model, the cake is represented by the interval $[0, 1]$. Each player $i \in [n]$ (where $[n] = \{1, \ldots, n\}$) has a non-atomic (additive) measure $v_i(\cdot)$, mapping each measurable subset of $[0, 1]$ to its value according to player i. For most of this work, we are only interested in a value of *intervals* in $[0, 1]$, and thus simply write $v_i(a, b)$ for the value of the interval between a and b. (Note that since v_i is non-atomic, single points have zero value, and we need not worry about the boundary points a and b themselves.)

We also assume, as common in the cake-cutting literature, that the valuations are *normalized*, i.e. that $v_i(0, 1) = 1$ for every player i. However, our results hold (with small

modifications to the algorithms or complexity) for arbitrary valuations as well.

Social Welfare Functions. We consider two prominent social welfare functions, whose aim is to measure how good each division is for the *whole society*. Let x be a division (to be formally defined shortly); we write $u_i(x)$ to express the value player i obtains from the piece she receives in x. The *utilitarian welfare* is defined as the sum of utilities, and we denote $u(x) = \sum_{i \in [n]} u_i(x)$. The *egalitarian welfare* is defined as the utility of the worst-off player, and we denote $eg(x) = \min_{i \in [n]} u_i(x)$.

Connected Divisions. In this work, we focus on divisions in which every player gets a (disjoint) *single interval* of the cake. Formally, a connected division of the cake $[0, 1]$ between n players can be defined as a vector $x = (x_1, \ldots, x_{n-1}, \pi) \in [0, 1]^{n-1} \times S_n$ (where S_n is the set of all the permutations of $[n]$), having $x_1 \leq x_2 \leq \cdots \leq x_{n-1}$. This is interpreted as making $n - 1$ cuts in positions $x_1, \ldots, x_{n-1}$, and allocating the n resulting intervals to the players in the order determined by the permutation π. Note that the space X of all such divisions is compact; in addition, both utilitarian and egalitarian welfare functions are continuous in X (as the players' valuation functions are all non-atomic). Therefore, for each of these welfare functions there exists a division that maximizes the welfare.

Our main problem is thus the following: *given the players' valuations, what is the (connected) division that maximizes welfare?* Since the two welfare functions considered here obtain maxima in the divisions space, the problem is indeed well-defined. For the analysis of these problems, it is useful to consider their decision versions, defined as follows.

CONNECTED UTILITARIAN OPTIMUM (CUO)
Instance: A set $\{v_i\}_{i=1}^n$ of non-atomic measures on $[0, 1]$, and a bound B.
Problem: Does there exist a connected division x having $u(x) \geq B$?

CONNECTED EGALITARIAN OPTIMUM (CEO)
Instance: A set $\{v_i\}_{i=1}^n$ of non-atomic measures on $[0, 1]$, and a bound B.
Problem: Does there exist a connected division x having $eg(x) \geq B$?

Complexity and Input Models. In order to analyze the complexity of our problems, we must first define how the input is represented. In most of the cake cutting literature, the mechanism is not explicitly given the players' valuation functions; instead, it can *query* the players on their valuations (see e.g. [EP84, RW98, Str08]). Typically, two types of queries are allowed. In the first, a player i is given points $0 \leq a \leq b \leq 1$ and is required to return the value $v_i(a, b)$. In the second type of query, a player i is given a point $a \in [0, 1]$ and a value x and is required to return a point b such that $v_i(a, b) = x$; we denote this by $v_i^{-1}(a, x)$.[1]

In contrast, some more recent works (e.g. [CLPP10, CLPP11, BCH+12]) consider a model in which the players give complete descriptions of their valuations to the mechanism. In this case, it is usually assumed that the functions have some simple structure, so they can be represented succinctly. Specifically, for each player i, let $\nu_i : [0, 1] \to [0, \infty)$ be a *value density function*, such that

$$v_i(X) = \int_X \nu_i(x)dx$$

for every measurable subset $X \in [0, 1]$. Following [CLPP10], we say that a valuation function $v_i(\cdot)$ is *piecewise-constant* if its value density function $\nu_i(\cdot)$ is a step function, i.e. if $[0, 1]$

---

[1]Note that using only one type of query it is possible to give approximate answers (in polynomial time) to queries of the other type using binary search.

can be partitioned into a *finite* number of intervals such that ν_i is constant on each interval. If, in addition, there is some constant c_i such that $\nu_i(\cdot)$ can only attain the values 0 or c_i, we say that $v_i(\cdot)$ is *piecewise-uniform*.[2] Any piecewise-constant valuation function $v_i(\cdot)$ can be therefore represented by a finite set of subintervals of $[0, 1]$ together with the value ν_i attains in each interval.

Our hardness results show that both of the decision problems above are computationally hard, even when the valuation functions are of the simplest type—piecewise-uniform—and are given explicitly to the mechanism. In contrast, our positive algorithmic results hold also for the more general oracle model. The complexity of our algorithms in this case depends on the number of players n and additionally on a precision parameter ϵ.

The Discrete Variants. A convenient preprocessing step in our algorithms will be reducing our problems into ones that are purely combinatorial. More precisely, we consider discrete analogues of the problems, where one is additionally given a set of points in $[0, 1]$, and is only allowed to make cuts at points from this set (and not anywhere in $[0, 1]$). An alternative interpretation is to consider, instead of a continuous cake, a *sequence of indivisible items*; a connected division in this setting gives each player a *consecutive subsequence* of these items. The discrete variants of our problems are defined as follows:

DISCRETE CONNECTED UTILITARIAN OPTIMUM (DISCRETE-CUO)
Instance: A sequence $A = (a_1, \ldots, a_m)$ of items, a set $\{v_i\}_{i=1}^n$ of valuation functions of the form $v_i : A \to \mathbb{R}^+$, and a bound B.
Problem: Does there exist a connected division x having $u(x) \geq B$?

DISCRETE CONNECTED EGALITARIAN OPTIMUM (DISCRETE-CEO)
Instance: A sequence $A = (a_1, \ldots, a_m)$ of items, a set $\{v_i\}_{i=1}^n$ of valuation functions of the form $v_i : A \to \mathbb{R}^+$, and a bound B.
Problem: Does there exist a connected division x having $eg(x) \geq B$?

Our hardness results apply to these "cleaner" problems as well. We note that if we drop the contiguity requirement, allowing players to get any disjoint subsets of A, maximizing utilitarian welfare becomes trivial (give each item to the player who values it the most). In contrast, maximizing egalitarian welfare (in the discrete setting with non-connected pieces) is known to be a hard problem [BD05] and has been studied extensively (e.g. [AS07, CCK09]).

3 Approximation Algorithms

In this section we present algorithms that return a division that is guaranteed not to be too far from the social optimum. Throughout this section we assume that the algorithms operate in the (more-general) oracle model. We note that if the valuation functions are given explicitly, and are simple enough (in particular, if they are piecewise-constant), the answer to each oracle query can be computed in time polynomial in the input size.

3.1 The Discretization Procedure

As we have previously mentioned, it is often useful to reduce the continuous cake into a sequence of discrete items. We now show that this can indeed be done in a time-efficient manner, and without too much harm to the maximum obtainable welfare.

---

[2]Note that in this case the constant c_i is uniquely determined by the total fraction of $[0, 1]$ in which $\nu_i(x) \neq 0$, since we require that the valuation of the entire cake should be 1.

The *Discretization Procedure* receives a cake instance and a parameter ϵ, and produces a set of cut positions that partition the cake into a set of items. We start with the set $C = \{0\}$ of cut points. At each step, let a be the position of the last (rightmost) cut in C. The procedure asks each player i for the leftmost point b_i such that the $v_i(a, b_i) = \epsilon$; it then adds the leftmost of these points to C, and repeats the process. When $v_i(a, 1) \leq \epsilon$ for all players i, the procedure adds the point 1 to C, and halts.

Note that the set of cuts C produced by the algorithm induces a sequence of items. Specifically, let $0 = c_0 < c_1 < \cdots < c_m = 1$ be the cut points in C; then, for each $1 \leq j \leq m$ create an item a_j with value $v_i(a_j) = v_i(c_{j-1}, c_j)$ for player $i \in [n]$.

The following lemma, whose proof we omit due to space constraints, establishes that the set C can be computed efficiently, and that we do not lose much utilitarian welfare by restricting our cuts positions to C. A similar claim also holds for egalitarian welfare.

Lemma 1. *Let $\{v_i(\cdot)\}_{i \in [n]}$ be a cake instance with n players, and consider some precision parameter ϵ. Then:*

1. *The discretization procedure terminates on this instance in time $O(n^2/\epsilon)$.*

2. *Let x be a division of the original cake; then there exists a division y making cuts only at points in the set C returned by the procedure, and having $u(y) \geq u(x) - (n-1)\epsilon$.*

3.2 Approximating the Utilitarian Welfare

We now present an approximation algorithm for the problem of maximizing utilitarian welfare; the approximation ratio achieved by our algorithm is $8(1 + (n-1)\epsilon)$, where ϵ is a precision parameter, and the running time of the algorithm is polynomial in n and in $1/\epsilon$. As a first step, the algorithm uses the Discretization Procedure to obtains a set A of m discrete items. We now describe how to approximate the optimal utilitarian welfare for this new instance. The algorithm returns a set $\{(s_i, t_i)\}_{i \in [n]}$, where s_i is the beginning index of i's piece, and t_i is its end index. We also use the notation (s, t) to refer to the consecutive sequence of items $\{s, s+1, \ldots, t-1, t\}$; hence, e.g. $v_i(s, t) = \sum_{j=s}^{t} v_i(j)$.

Algorithm 1: Discrete Utilitarian Welfare Approximation

> **Data**: For each player $i \in [n]$ a vector of valuations $v_i : [m] \to \mathbb{R}^+$.
> **begin**
> > $\forall i \in [n] : s_i \longleftarrow 0$, $t_i \longleftarrow 0$
> > **for** $t = 1, \ldots, m$ **do**
> > > **while** $\max_{k \in [n], s \leq t} \left\{ v_k(s, t) - 2\left(v_k(s_k, t_k) + V_{-k}(s, t) \right) \right\} \geq 0$
> > > **do**
> > > > $k', s' \longleftarrow$ arguments maximizing the expression
> > > > $s_{k'} \longleftarrow s'$, $t_{k'} \longleftarrow t$
> > > > $(s_i, t_i) \longleftarrow (0, 0)$ for all i with $s_i \geq s'$
> > > > $t_i \longleftarrow s' - 1$ for i with $s_i < s' \leq t_i$
> >
> > **return** $\{(s_i, t_i)\}_{i \in [n]}$

Our algorithm for the discretized instance works iteratively, where in the t-th iteration it finds a good division for the first t items. We begin with the trivial null allocation of 0 items. Assuming that we have a good allocation for the first $t - 1$ items, and for all $s \leq t$ and $k \in [n]$, we consider the *cost* of giving items s through t to player k. This cost

is comprised of two components. The first component is the value of a piece (s_k, t_k) that player k may currently own, and has to give up in order to get the new piece (s, t). The second component is the sum of values that the other players to which the items s through t are assigned obtain from these items. We denote this second component by $V_{-k}(s, t)$. We only give the segment (s, t) to player k if her total value $v_k(s, t)$ for this segment is at least twice the cost of giving her this segment. We continue trying to find a player k' and a segment (s', t) ending at item t whose value exceeds twice the cost, and make changes until there are no such player and segment, at which point we move on to the next item $t + 1$.

Observe that in the algorithm, each interval (s, t) can be given to player i at most once; this immediately implies that the running time of the algorithm is polynomial in the number of players n and number of items m. For analyzing the approximation ratio of the algorithm, we use indicator variables x_i^j, for $i \in [n]$ and $j \in [m]$. At each step in the algorithm, we will have $x_i^j = 1$ if and only if player i owned the item j at some point until now.

Lemma 2. *At any iteration t of the above algorithm, we have*

$$\sum_{i \in [n]} v_i(s_i, t_i) \leq \sum_{i \in [n]} \sum_{j \in [m]} x_i^j \cdot v_i(j) \leq 2 \cdot \sum_{i \in [n]} v_i(s_i, t_i)$$

(where the values are as in the end of the t-th iteration).

Proof. The first inequality trivially holds, and we prove the second by induction on t. The second inequality clearly holds at the beginning of the step $t = 1$; we show that if it holds at the beginning of some step t, then it must still hold at the end of this step.

At the beginning of the t-th step, item t is unallocated. If the while loop was not executed even once in this iteration, none of the expressions $\sum_{i \in [n]} \sum_{j \in [m]} x_j^i \cdot v_i(j)$ and $\sum_{i \in [n]} v_i(s_i, t_i)$ have changed, and the claim still holds. Otherwise, consider some iteration of the while loop. In such an iteration, the increase in $\sum_{i \in [n]} \sum_{j \in [m]} x_i^j \cdot v_i(j)$ is upper-bounded by $v_{k'}(s', t)$. The expression $\sum_{i \in [n]} v_i(s_i, t_i)$ also gains $v_{k'}(s', t)$, but in addition loses $v_k(s_k, t_k) + V_{-k}(s, t)$; however, the while loop condition ensures that $v_{k'}(s', t) - \big(v_k(s_k, t_k) + V_{-k}(s, t) \big) \geq \frac{1}{2} \cdot v_{k'}(s', t)$. Therefore, the increase to the right-hand side of the inequality is at least as large as that of the left-hand side, and the inequality is maintained. Since this holds for every iteration of the while loop, this still holds at the end of step t, as required. $\qquad\square$

Theorem 1. *Algorithm 2 returns an 8-approximation of the discrete utilitarian optimum.*

Proof. Fix a discrete cake instance. Let $\big\{ (s_i^A, t_i^A) \big\}_{i \in [n]}$ be the final output of Algorithm 1 on this instance, and let $\big\{ (s_i^*, t_i^*) \big\}_{i \in [n]}$ be the optimal division for this instance. Denote by $OPT = \sum_{i \in [n]} v_i(s_i^*, t_i^*)$ the utilitarian welfare achieved by the optimal division.

For every player k, consider the iteration t_k^*, in which the rightmost item given to k in the optimal division was first considered. Let (s_k', t_k') be the segment given to player k at the end of this iteration. When iteration t_k^* ends, it has to be that

$$v_k(s_k^*, t_k^*) \leq 2\big(v_k(s_k', t_k') + V_{-k}(s_k^*, t_k^*) \big)$$

(where $V_{-k}(s_k^*, t_k^*)$ is with respect to the division set by the algorithm at this point). Note that $v_k(s_k', t_k') = \sum_{j=s_k'}^{t_k'} x_j^k \cdot v_k(j)$ and that $V_{-k}(s_k^*, t_k^*) \leq \sum_{j=s_k^*}^{t_k^*} \sum_{i \neq k} x_j^i \cdot v_i(j)$. Combining

all this, we get

$$OPT = \sum_{k\in[n]} v_k(s_k^*, t_k^*) \leq \sum_{k\in[n]} 2 \cdot \Big(\sum_{j=s_k'}^{t_k'} x_j^k \cdot v_k(j) + \sum_{j=s_k^*}^{t_k^*} \sum_{i\neq k} x_j^i \cdot v_i(j) \Big)$$

$$= 2 \cdot \Big(\sum_{k\in[n]} \sum_{j=s_k'}^{t_k'} x_j^k \cdot v_k(j) + \sum_{k\in[n]} \sum_{j=s_k^*}^{t_k^*} \sum_{i\neq k} x_j^i \cdot v_i(j) \Big)$$

$$\leq 2 \cdot \Big(\sum_{k\in[n]} \sum_{j\in[m]} x_j^k \cdot v_k(j) + \sum_{k\in[n]} \sum_{j\in[m]} x_j^k \cdot v_k(j) \Big)$$

$$= 4 \cdot \sum_{k\in[n]} \sum_{j\in[m]} x_j^k \cdot v_k(j) \leq 8 \cdot \sum_{i\in[n]} v_i(s_i^A, t_i^A) \ .$$

The second inequality holds since for every $k \neq k'$ the segments (s_k^*, t_k^*) and $(s_{k'}^*, t_{k'}^*)$ are disjoint, as $\big\{(s_i^*, t_i^*)\big\}_{i\in[n]}$ is a division. The last inequality follows from Lemma 2. $\qquad\square$

Combining the guarantees for the Discretization Procedure and for Algorithm 1 we get:

Corollary 2. *For every $\epsilon > 0$, it is possible to find a division achieving utilitarian welfare within $8\big(1 + (n-1)\epsilon\big)$ of the optimum in time polynomial in n and $1/\epsilon$.*

3.3 Fixed-Parameter Tractable Approximations

Suppose that we have a relatively small number of players n, but that the social efficiency of the division is of much importance. We show that divisions that are within a factor of $1 + \epsilon$ of the social optimum (for both utilitarian and egalitarian welfare) can be computed in time exponential in the number of players, but polynomial in $\frac{1}{\epsilon}$.[3] Using the terminology of the theory of Parametrized Complexity [DF99] we say that these approximations are *fixed-parameter tractable*. Both of these algorithms are based on dynamic programming; the full proofs can be found in the full version of the paper.

Theorem 3. *For every $\epsilon > 0$, it is possible to find a division achieving utilitarian welfare within $1 + \epsilon$ of the optimum in time $2^n \cdot poly(n, \frac{1}{\epsilon})$.*

Theorem 4. *For every $\epsilon > 0$, it is possible to find a division achieving egalitarian welfare within $1 + \epsilon$ of the optimum in time $2^n \cdot n \cdot \log_2 \big(\frac{n}{\epsilon}\big)$.*

4 Hardness

We show that all of the four problems defined in Section 2 are NP-complete in the strong sense. Note that membership in NP is straightforward, as a division achieving the required welfare can serve as a witness for that instance; we thus concentrate on proving hardness.

We prove that CEO is strongly NP-complete and hard to approximate to a factor of $2-\epsilon$ for any $\epsilon > 0$, using a reduction from the classic problem of 3DM [GJ79]. In this problem, one is given three sets X, Y, Z of cardinality n each, as well as a set $E \subseteq X \times Y \times Z$, and needs to determine if there exists a subset $E' \subseteq E$ of cardinality n that covers X, Y and Z.

Our reduction borrows its main ideas from the proof of Bezáková and Dani [BD05] for non-connected divisions in the discrete setting, which itself uses ideas from Lenstra, Shmoys and Tardos [LST90]. However, the adjustment to the continuous setting with connected divisions is somewhat intricate and needs to be done carefully.

---

[3]Recall that we assume the oracle model; if the valuation functions are given explicitly, we also have polynomial dependence on the size of the input.

Theorem 5. CEO *and* DISCRETE-CEO *are NP-complete in the strong sense. Furthermore, for every $\epsilon > 0$ there is no $(2 - \epsilon)$ approximation for either of the problems, unless P=NP.*

This holds even if the valuation functions of the players are piecewise-uniform, and are given explicitly to the algorithm.

Proof. We show a polynomial-time reduction from 3DM to CEO. Let X, Y, Z and $E \subseteq X \times Y \times Z$ be an input to 3DM. We construct a set of piecewise-constant valuations and a bound B as an input for CEO; this instance can be transformed into an equivalent one with piecewise-uniform valuations.

For convenience, we take the cake to be the interval $[0, 2|E|]$ rather than $[0, 1]$. We will think of the cake as being sectioned into $|E|$ "sections" of length 2, where the right half of each section is used for separation from the next section.[4] The set of players we create has players of three types: "triplet players", "ground sets players" and "separation players". In what follows we describe the valuation functions of all the players, by their type; for the bound, we set $B = \frac{1}{|E|}$.

- **Triplet Players:** We create a player for every $z \in Z$. For each $e_i \in E$ such that z appears in the triplet e_i, the player created for z has value of $\frac{1}{2|E|}$ for each of the intervals $\left(2(i-1), 2(i-1) + \frac{1}{4}\right)$ and $\left(2(i-1) + \frac{3}{4}, 2(i-1) + 1\right)$ in the left half of the i-th section.

  Denote by m_z the number of such triplets e_i in E. To keep the value of the entire cake at 1 for each player, we will divide the missing value $1 - \frac{m_z}{|E|}$ between the right halves of all sections. Specifically, player z will additionally have value $\frac{|E| - m_z}{2|E|^2}$ for every interval $\left(2(j-1) + \frac{6}{5}, 2(j-1) + \frac{7}{5}\right)$ and $\left(2(j-1) + \frac{8}{5}, 2(j-1) + \frac{9}{5}\right)$, for all $1 \le j \le |E|$.

- **Ground Sets Players:** For $x \in X$, let m_x be the number of triplets in E in which x appears. We create $m_x - 1$ identical players for x. For every $e_i \in |E|$ such that x appears in e_i, all of x's players will have valuation of $\frac{1}{|E|}$ for the interval $\left(2(i-1) + \frac{1}{4}, 2(i-1) + \frac{1}{2}\right)$ in the left half of the i-th section. Again, in order to complement these valuations to 1, they will also assign a value of $\frac{|E| - m_x}{2|E|^2}$ for each of the intervals $\left(2(j-1) + \frac{6}{5}, 2(j-1) + \frac{7}{5}\right)$ and $\left(2(j-1) + \frac{8}{5}, 2(j-1) + \frac{9}{5}\right)$, for all $1 \le j \le |E|$.

  We similarly create $m_y - 1$ identical players for every $y \in Y$. For each $e_i \in E$ in which y appears we have these players give value of $\frac{1}{|E|}$ to the interval $\left(2(i-1) + \frac{1}{2}, 2(i-1) + \frac{3}{4}\right)$, and complement this by giving value of $\frac{|E| - m_y}{2|E|^2}$ to each of the intervals $\left(2(j-1) + \frac{6}{5}, 2(j-1) + \frac{7}{5}\right)$ and $\left(2(j-1) + \frac{8}{5}, 2(j-1) + \frac{9}{5}\right)$, for all $1 \le j \le |E|$.

- **Separation Players:** We finally create $3|E|$ separation players. For every segment $1 \le i \le |E|$ we have a player s_{3i-2} have valuation of 1 for the interval $\left(2(j-1) + 1, 3(j-1) + \frac{6}{5}\right)$, another player s_{3i-1} have valuation 1 for $\left(2(i-1) + \frac{7}{5}, 2(i-1) + \frac{8}{5}\right)$, and a third player s_{3i} have valuation 1 for $\left(2(i-1) + \frac{9}{5}, 2(j-1) + 2\right)$.

Figure 1 illustrates the structure of the preferences in one segment. In this example, we consider some triplet $e_i = (x_j, y_k, z_\ell) \in E$, and show the section of the cake created for it, with the preferences of the players who desire some piece of it.

It is straightforward to observe that the construction above can be carried out in polynomial time. Also, all the numbers created in this instance can be represented by a number of bits logarithmic in the input size.

---

[4]Indeed, the last section needs not have this "separation half"; however, we leave it there in order to treat it identically to all the other sections.
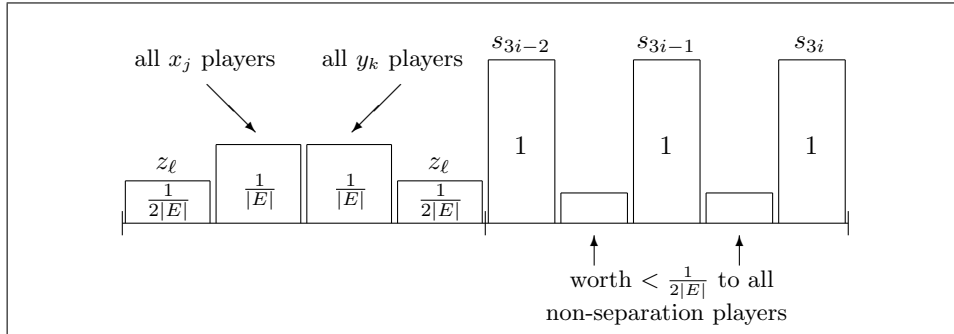
Figure 1: The valuations of the players for the section created for $e_i = (x_j, y_k, z_\ell) \in E$. Note that there are $m_{x_j} - 1$ identical players for x_j and $m_{y_k} - 1$ identical players for y_k.

Due to space constraints, we defer the correctness proof for this construction, as well as the adjustment for piecewise-uniform valuations, to the full version of the paper.

The proof for DISCRETE-CEO is analogous, and can easily be obtained by a straightforward partitioning of the cake created in the reduction into discrete indivisible chunks. □

We use a reduction from DISCRETE-CEO to prove the hardness of maximizing utilitarian welfare. The proof is again deferred to the full version due to space constraints.

Theorem 6. CUO *and* DISCRETE-CUO *are NP-complete in the strong sense.*

This holds even if the valuation functions of the players are piecewise-uniform, and are given explicitly to the algorithm.

The strong NP-hardness of CUO and DISCRETE-CUO implies the following corollary:

Corollary 7. *There is no FPTAS for either* CUO *nor* DISCRETE-CUO*.*

5 Welfare Maximization with Non-Connected pieces

In this section we analyze the problem of welfare maximization when each player may get a *collection* of intervals. We first show that if the valuation functions are piecewise-constant and are given explicitly to the algorithm, the problem can be easily solved in polynomial time using a linear program almost identical to the one used by Cohler et al. [CLPP11]; the details of the proof are can be found in the full version of the paper.

Theorem 8. *Given a set of n piecewise-constant valuation functions (i.e. for each $i \in [n]$ the list of intervals in which the value density function attains different values, along with the value for each such interval), it is possible to find a division maximizing the utilitarian (resp. egalitarian) welfare in polynomial time.*

In contrast to this positive result, it turns out that maximizing welfare is *impossible* if instead of receiving the explicit functions, we only get oracle access to the valuations. In particular, we show that no deterministic algorithm (even super-polynomial) can find a division approximating the utilitarian or egalitarian optimum by a factor smaller than n. Note that this bound is tight, as every *proportional* division[5] approximates utilitarian and egalitarian welfare by at least n, and many algorithms for finding proportional divisions do exist in the queries model (see, e.g. [RW98] for a survey).

---

[5]Recall that a division is said to be proportional if it gives each player what she considers to be at least $1/n$ of the total value of the cake.

Theorem 9. *For any $\epsilon > 0$, no deterministic algorithm working in the oracle input model can approximate utilitarian or egalitarian welfare to a factor of $n - \epsilon$, when non-connected pieces are allowed.*

Proof sketch. We discuss utilitarian welfare; the arguments for egalitarian welfare are analoguous. Let A be a deterministic cake division algorithm working in the oracle input model, and fix some $n \in \mathbb{N}$ and $\epsilon > 0$. Consider the operation of the algorithm on the set of preferences in which all players value the entire cake uniformly. In this case, the utilitarian welfare obtained cannot exceed 1. We will now show that for any $\epsilon' > 0$ we can construct a different set of preferences on which the algorithm outputs the same division (with the same welfare), but for which there exists a division achieving utilitarian welfare of $(1 - \epsilon')n$. The theorem will follow by choosing $\epsilon' = \epsilon/n$.

Let $0 = p_0 < p_1 < \ldots < p_{k-1} < p_k = 1$ be the set of (distinct) points that appear in the operation of the algorithm on the input above. I.e. $\{p_i\}_{i=0}^k$ is the set of all points a, b for which the algorithm makes a query $v_i(a, b)$ or receives an answer $b = v_i^{-1}(a, x)$, and all the points c in which the algorithm makes cuts in its output division. We create a new instance in which the valuations in the interval between two each consecutive such points (p_j, p_{j+1}) are "rearranged". The value of this interval in the original instance, as well as in the new instance, is $\ell_j = p_{j+1} - p_j$. We divide this interval into $n + 1$ "slivers": the i-th sliver ($1 \leq i \leq n$) is worth $\ell_j - \frac{\epsilon'}{k}$ to player i, and zero to everyone else. The $n + 1$-st sliver of the interval is worth $\frac{\epsilon'}{k}$ for all the players.

It is straightforward to observe that the operation of the algorithm A is identical on the old and new instances, as we constructed the new valuations so that the answer to every query asked in the operation of A is identical in the two instances. This implies that A returns the same division for both instances, and the utilitarian welfare of this division is 1 in both of them. However, in the new instance, any division that allocates every sliver desired only by one player to this player, achieves utilitarian welfare $> (1 - \epsilon')n$. $\square$

6 Open Problems

In this work we have taken the first steps in studying the problem of maximizing welfare in cake cutting with connected pieces. Many interesting problems related to this problem remain open. First and foremost, we believe that it should be possible to obtain a reasonable approximation for the problem of maximizing the egalitarian welfare. (We do have non-trivial algorithms that achieve linear-factor approximations, but we conjecture that better algorithms can be found.) We also conjecture that the approximation ratio for maximizing utilitarian welfare can be improved; it may also be interesting to see if stronger inapproximability results can be shown. Other interesting extensions include:

- *Strategic Behavior:* One implicit assumption in our work was that we have access to the (true) valuations of the players. In reality, the players may have incentive to lie about their valuations. Guo and Conitzer [GC10] and Han et al. [HSTZ11] have considered this problem for a somewhat different setting; the question of what can be achieved truthfully in our setting is still open.

- *2-Dimensional Cake:* The cake cutting literature has generally assumed a one-dimensional cake; indeed, for the purpose of maintaining fairness, which was its main focus, a 2-dimensional cake can be simply "projected" into one dimension, and divided fairly according to the projection. However, this may result in a significant loss of welfare. Therefore, maximizing welfare in allocation of 2-dimensional cakes may require completely different tools and techniques.

References

[AAWY98] Noga Alon, Yossi Azar, Gerhard J. Woeginger, and Tal Yadid. Approximation schemes for scheduling on parallel machines. *Journal of Scheduling*, 1:55–66, 1998.

[AD10] Yonatan Aumann and Yair Dombb. The efficiency of fair division with connected pieces. In *WINE*, pages 26–37, 2010.

[AS07] Arash Asadpour and Amin Saberi. An approximation algorithm for max-min fair allocation of indivisible goods. In *STOC*, pages 114–121, 2007.

[BCH+12] Xiaohui Bei, Ning Chen, Xia Hua, Biaoshuai Tao, and Endong Yang. Optimal proportional cake cutting with connected pieces. In *AAAI*, 2012.

[BD05] Ivona Bezáková and Varsha Dani. Allocating indivisible goods. *SIGecom Exchanges*, 5(3):11–18, 2005.

[BS06] Nikhil Bansal and Maxim Sviridenko. The santa claus problem. In *STOC*, pages 31–40, 2006.

[BT95] Steven J. Brams and Alan D. Taylor. An envy-free cake division protocol. *The American Mathematical Monthly*, 102(1):9–18, 1995.

[BT96] Steven J. Brams and Alan D. Taylor. *Fair Division: From cake cutting to dispute resolution.* Cambridge University Press, New York, NY, USA, 1996.

[CCK09] Deeparnab Chakrabarty, Julia Chuzhoy, and Sanjeev Khanna. On allocating goods to maximize fairness. In *FOCS*, pages 107–116, 2009.

[CKKK09] Ioannis Caragiannis, Christos Kaklamanis, Panagiotis Kanellopoulos, and Maria Kyropoulou. The efficiency of fair division. In *WINE*, pages 475–482, 2009.

[CLPP10] Yiling Chen, John Lai, David C. Parkes, and Ariel D. Procaccia. Truth, justice, and cake cutting. In *AAAI*, 2010.

[CLPP11] Yuga J. Cohler, John Lai, David C. Parkes, and Ariel Procaccia. Optimal envy-free cake cutting. In *AAAI-11: Proc. 25th AAAI Conference on Artificial Intelligence*, 2011.

[DF99] R. G. Downey and M. R. Fellows. *Parameterized Complexity.* Springer, 1999.

[DS61] L. E. Dubins and E. H. Spanier. How to cut a cake fairly. *The American Mathematical Monthly*, 68(1):1–17, Jan 1961.

[EP84] S. Even and A. Paz. A note on cake cutting. *Discrete Applied Mathematics*, 7(3):285 – 296, 1984.

[Fei08] Uriel Feige. On allocations that maximize fairness. In *SODA*, pages 287–293, 2008.

[GC10] Mingyu Guo and Vincent Conitzer. Strategy-proof allocation of multiple items between two agents without payments or priors. In *AAMAS*, pages 881–888, 2010.

[GJ79]   M. R. Garey and D. S. Johnson. *Computers and intractability: A guide to the theory of NP-completeness*. W. H. Freeman, 1979.

[HSTZ11] Li Han, Chunzhi Su, Linpeng Tang, and Hongyang Zhang. On strategy-proof allocation without payments or priors. In *WINE*, pages 182–193, 2011.

[LMMS04] Richard J. Lipton, Evangelos Markakis, Elchanan Mossel, and Amin Saberi. On approximately fair allocations of indivisible goods. In *ACM Conference on Electronic Commerce*, pages 125–131, 2004.

[LST90]   Jan Karel Lenstra, David B. Shmoys, and Éva Tardos. Approximation algorithms for scheduling unrelated parallel machines. *Math. Program.*, 46:259–271, 1990.

[Pro09]   Ariel D. Procaccia. Thou shalt covet thy neighbor's cake. In *IJCAI*, pages 239–244, 2009.

[RW98]   Jack Robertson and William Webb. *Cake-cutting algorithms: Be fair if you can.* A K Peters, Ltd., Natick, MA, USA, 1998.

[Ste49]   H. Steinhaus. Sur la division pragmatique. *Econometrica*, 17(Supplement: Report of the Washington Meeting):315–319, Jul 1949.

[Str80]   Walter Stromquist. How to cut a cake fairly. *The American Mathematical Monthly*, 87(8):640–644, 1980.

[Str08]   Walter Stromquist. Envy-free cake divisions cannot be found by finite protocols. *Electronic Journal of Combinatorics*, 15(1), Jan 2008.

[SW03]   Jiri Sgall and Gerhard J. Woeginger. A lower bound for cake cutting. In *ESA*, pages 459–469, 2003.

[Woe97]   Gerhard J. Woeginger. A polynomial-time approximation scheme for maximizing the minimum machine completion time. *Operations Research Letters*, 20(4):149 – 154, 1997.

[Ziv11]   Roie Zivan. Can trust increase the efficiency of cake cutting algorithms? In *AAMAS*, pages 1145–1146, 2011.

Yonatan Aumann
Department of Computer Science
Bar-Ilan University
Ramat Gan 52900, Israel
Email: `aumann@cs.biu.ac.il`

Yair Dombb
Department of Computer Science
Bar-Ilan University
Ramat Gan 52900, Israel
Email: `yair.biu@gmail.com`

Avinatan Hassidim
Department of Computer Science
Bar-Ilan University
Ramat Gan 52900, Israel
Email: `avinatanh@gmail.com`