DS 574 Algorithmic Mechanism Design	**Lecture #18**
Fall 2025	Prof. Kira Goldner

Cake Cutting: Divisible Goods

Properties of the Cut and Choose Protocol

Today, we'll focus primarily on fairness properties. Suppose two people need to split a heterogeneous and divisible good. The usual euphemism in the fair division literature is that of cutting a cake. More practically, the good could be an estate (e.g., in a divorce settlement) or processing time on a computer cluster (perhaps with some times of the day more valuable than others).

Why not just split the good 50/50? This makes sense when the good is homogeneous, but it's not clear what this means with a heterogeneous good. A player may value a part of the good much more than another, and different players can have different opinions about which parts are the most valuable.

Let's talk about the most famous protocol for two-person cake-cutting.

The Cut and Choose Protocol

1. Player 1 splits the good into two pieces A and B, such that the player's value for each is exactly half that of the entire good.
2. Player 2 picks whichever of A, B she likes better.

The description above is the intended behavior of the players in the protocol—we'll talk shortly about whether or not they are incentivized to follow this behavior. We've been led all our lives to believe that this is a "fair" protocol. But is it? How would we formally argue one way or the other?

Here's the formal model. The "cake" is the unit interval $[0, 1]$. Each player i has a valuation function v_i, which specifies the value $v_i(S)$ to i of a given subset S of the cake. We'll make the following two assumptions about each valuation function v_i:[1]

1. We normalize each v_i such that $v_i([0, 1]) = 1$, their value for the whole cake. This is more or less without loss of generality, by scaling.
2. v_i is additive on disjoint subsets. That is, if $A, B \subseteq [0, 1]$ are disjoint, then

$$v_i(A) + v_i(B) = v_i(A \cup B).$$

[1] Actually, we also need a "continuity" assumption for everything to make sense—e.g., in the cut and choose protocol, it's important that there exists a cut that makes player indifferent between the two pieces.

Is the cut and choose protocol strategyproof?

Is the cut and choose protocol guaranteed to produce a Pareto optimal solution (assuming both players behave as intended)?

Can we come up with a protocol that is both Pareto optimal and strategy proof for both players, or is there a fundamental impossibility result?

What about "fairness?" What are possible notions? Equally happy?

Definition 1. An allocation $A_1, A_2, \ldots, A_n$ of cake to n players is *proportional* if

for every player i.

Definition 2. An allocation $A_1, A_2, \ldots, A_n$ of cake to n players is *envy-free* if

for every pair i, j of players.

Do you notice a relationship between these? Protocols that satisfy one but not the other?

Beyond Two Players

The obvious next question is to ask about analogs of the cut and choose protocol that work with 3 or more players. We focus on envy-freeness and ignore strategyproofness issues.

$n = 3$ **players:** Selfridge and Conway (∼1960) independently designed the same envy-free protocol for this case, see exposition in [5].

- The third player cuts the cake into three pieces such that she has value $\frac{1}{3}$ for each of them.
- The first and second players then select their favorite pieces.
- If they select different pieces, then we are done (why?). Otherwise, if the first and second players both prefer the same piece, say the first (wlog):
  - Then, the second player is asked to trim off a subpiece of the first piece such that she is indifferent between the trimmed piece and her second-favorite of the original three pieces.
  - The protocol then recurses on the trimmings with the roles of the players swapped (with the second player cutting the trimmings into three pieces of equal value to her).

The protocol does not need to recurse further, because of the special structure of the recursively defined subproblem. In the worst case, the Selfridge-Conway protocol makes 5 cuts (the first two cuts by the third player, the trim by the second player, and two more cuts in the recursive call).

$n \geq 4$ **players:** In 1995, Brams and Taylor [3], gave a finite-but-unbounded protocol for computing an envy-free allocation with any number of players: for every $n \geq 4$ and T, there is a choice of $v_1, \ldots, v_n$ such that the protocol requires more than T steps to terminate.

In 2016, breakthrough results by Aziz and Mackenzie [2, 1] finally gave bounded protocols for envy-free protocols for $n \geq 4$ players. For $n = 2$, there are at most 203 cuts. For general n, it's a tower of 6 n's, meaning $n^{n^{\cdot^{\cdot^{n}}}}$ with 6 layers. As for lower bounds, the best known is $\Omega(n^2)$ [6]. Now there's a gap that's in need of narrowing!

Rent Division: Fair Division in Practice

One place where fair division protocols are used in practice is on `spliddit.org`, which has been used by tens of thousands of people. One of the problems that spliddit solves is the rent division problem, where there are n people, n rooms, and a rent of R. The goal is to assign people and rents to rooms, with one person per room and with the sum of rents equal to R, in the "best" way possible.

- Assume that each person i has a value v_{ij} for each room j.
- Normalize i's values so that their sum for all rooms is $\sum_j v_{ij} = R$.
- People have quasi-linear utility for their room minus rent paid for their room.[2]

[2]Rent division isn't really a special case of cake cutting, since the rooms are indivisible.

A solution to a rent division problem is envy-free if

$$v_{iM(i)} - p_{M(i)} \geq v_{iM(j)} - p_{M(j)} \tag{1}$$

for every pair i, j of players, where $M(i)$ denotes the room to which i is assigned and p_j denotes the rent assigned to the room j. That is, no one wants to trade places with anyone else (where trading places means swapping both rooms *and* rents).

This is essentially looking for a Walrasian Equilibrium, except that we have the added constraint that the prices must sum to the total rent.

The good news is that an envy-free solution is guaranteed to exist, and that one can be computed efficiently.

The bad news is that there can be many envy-free solutions, and not all of them are reasonable. For example, suppose there are two players and two rooms, that the total rent R is 1000, and that the first player only wants the first room ($v_{11} = 1000$ and $v_{12} = 0$) while the second player only wants the second room ($v_{21} = 0$ and $v_{22} = 1000$). What are envy free allocations and prices?

The upshot is that we need a method for selecting one out of the many envy-free solutions. One can imagine several ways of doing this; here's what happens on spliddit (given v_{ij}'s and R):

1. Choose an allocation (room assignment M) to maximize welfare ($\sum_i v_{iM(i)}$).
2. Set the room rents so that envy-freeness (1) holds, and subject to this, maximize the minimum utility:

$$\max_p \left(\min_{i=1}^{n} (v_{iM(i)} - p_{M(i)}) \right).$$

Algorithmically, they just solve this by computing a maximum-weight bipartite matching via linear programming. See [4] for further details.

Fair Division of Indivisible Goods

We now move on to indivisible goods, which is actually a much harder setting to achieve the same guarantees for. To see this, consider the example of two players and a single indivisible good.

We reiterate the model for this setting. Again, we will normalize valuations for each player such that if they receive *all of the goods*, their value is 1.

Definition 3. We have a set $N = [n]$ of n players and a set $\mathcal{M}$ of m goods. Each agent $i \in N$ has a monotone valuation function $v_i : 2^M \to \mathbb{R}_+$. We assume the valuations are normalized, i.e. $v_i(\emptyset) = 0,\ v_i(\mathcal{M}) = 1$ for all $i \in N$.

In many cases, we will also restrict our attention to the case of additive valuations as they are much easier to work with. A function v_i is said to be additive if it can be written as:

$$v_i(S) = \sum_{j \in S} v_i(j).$$

Now, that we have defined an instance of the fair division problem, let us discuss what we wish to achieve: a fair division. We begin by defining the notion of an allocation.

Definition 4. Given an instance of the fair division problem, we define an allocation $A = (A_1, \ldots, A_n)$ as a tuple of disjoint sets $A_i \subseteq \mathcal{M}$, i.e. $A_i \cap A_j = \emptyset$ for all $i \neq j$. We call the allocation a *partition* if the allocation allocates all the goods, i.e.

$$\bigcup_{i=1}^{n} A_i = \mathcal{M}.$$

Notions of Envy Freeness

Our definition of envy-freeness (EF) is the same as before. Since we have already seen that it is not always possible to achieve an EF allocation, we relax this condition and define two lesser notions of envy and thus fairness.

Definition 5. For an instance of the fair division problem $(N, \mathcal{M}, (v_1, \ldots, v_n))$ and an allocation $A = (A_1, \ldots, A_n)$:

1. We say that the allocation A is *envy-free up to one good* (EF1) if for all $i, j \in N$, we have that:
$$v_i(A_i) \geq v_i(A_j \setminus \{\alpha\}) \quad \text{for some } \alpha \in A_j.$$

2. We say that the allocation A is *envy-free up to any good* (EFX) if for all $i, j \in N$, we have that:
$$v_i(A_i) \geq v_i(A_j \setminus \{\alpha\}) \quad \text{for all } \alpha \in A_j.$$

When can we achieve an EF1 allocation? Let's try for $n = 2$ bidders, any m items, and additive valuations. [*Hint:* Try something like cut-and-choose.]

Example 1.

Player	A	B	C
1	10	9	2
2	1	8	4

The Round Robin Mechanism First, set an arbitrary order of the players; let us denote with 1 the first agent, with 2 the second agent, etc. Then, it proceeds as follows: until all the items are allocated, it goes through the players in the order $1, 2, \ldots, n$ in a cyclic manner. When it is the turn of agent i, he chooses the most valuable item (for him) which hasn't been allocated.

Theorem 1. *For fair division instance $(N, \mathcal{M}, v_1, \ldots, v_n)$ with additive valuations, there exists an EF1 allocation. Moreover, such an allocation is produced by the Round Robin Mechanism.*

Proof.

But, we claim that the Round Robin Mechanism can produce allocations which are intuitively very unfair. Consider the following example:

Acknowledgements

This lecture was developed in part using lecture notes by Moshe Babaioff and materials by Tim Roughgarden, and in particular, lectures from "CS269I: Incentives in Computer Science" [7].

References

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

[2] Haris Aziz and Simon Mackenzie. A discrete and bounded envy-free cake cutting protocol for four agents. In *Proceedings of the forty-eighth annual ACM symposium on Theory of Computing*, pages 454–464, 2016.

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

[4] Ya'akov Gal, Moshe Mash, Ariel D Procaccia, and Yair Zick. Which is the fairest (rent division) of them all? In *Proceedings of the 2016 ACM Conference on Economics and Computation*, pages 67–84, 2016.

[5] Erica Klarreich. How to cut cake fairly and finally eat it too. *Quanta Magazine*.

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

[7] Tim Roughgarden. CS269I: Incentives in Computer Science, 2016.