CSC2556

Lecture 6

Fair Division 1: Cake-Cutting Indivisible Goods

[Some illustrations due to: Ariel Procaccia]

Announcements

- Reminder
  - Project proposal due by March 3^{st} by 11:59PM
  - If you want to run your idea by me, this is a good time to approach me (email me and we'll setup a time to chat).

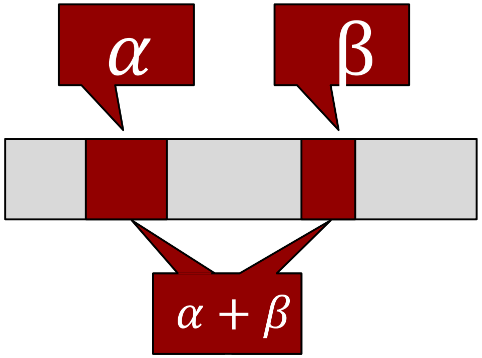
Fair Division

Cake-Cutting

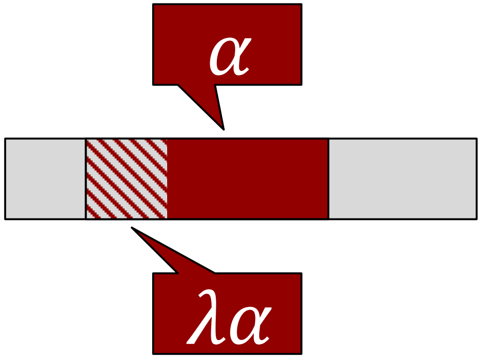
- A heterogeneous, divisible good
  - Heterogeneous: it may be valued differently by different individuals
  - Divisible: we can share/divide it between individuals
- Represented as $[0,1]$
  - Almost without loss of generality
- Set of players $N = \{1, \ldots, n\}$
- Piece of cake $X \subseteq [0,1]$
  - A finite union of disjoint intervals

Agent Valuations

- Each player i has a valuation V_i that is very much like a probability distribution over $[0,1]$



- Additive: For $X \cap Y = \emptyset$, $V_i(X) + V_i(Y) = V_i(X \cup Y)$
- Normalized: $V_i([0,1]) = 1$
- Divisible: $\forall \lambda \in [0,1]$ and X, $\exists Y \subseteq X$ s.t. $V_i(Y) = \lambda V_i(X)$



Fairness Goals

- An allocation is a disjoint partition $A = (A_1, \dots, A_n)$ of the cake
- We desire the following fairness properties from our allocation A:

- Proportionality (Prop):

$$\forall i \in N: V_i(A_i) \geq \frac{1}{n}$$

- Envy-Freeness (EF):

$$\forall i, j \in N: V_i(A_i) \geq V_i(A_j)$$

Fairness Goals

- Prop: $\forall i \in N: V_i(A_i) \geq 1/n$
- EF: $\forall i, j \in N: V_i(A_i) \geq V_i(A_j)$

- Question: What is the relation between proportionality and EF?
  1. Prop $\Rightarrow$ EF
  2. EF $\Rightarrow$ Prop
  3. Equivalent
  4. Incomparable

CUT-AND-CHOOSE

- Algorithm for $n = 2$ players

- Player 1 divides the cake into two pieces X, Y s.t.
$$V_1(X) = V_1(Y) = 1/2$$

- Player 2 chooses the piece she prefers.

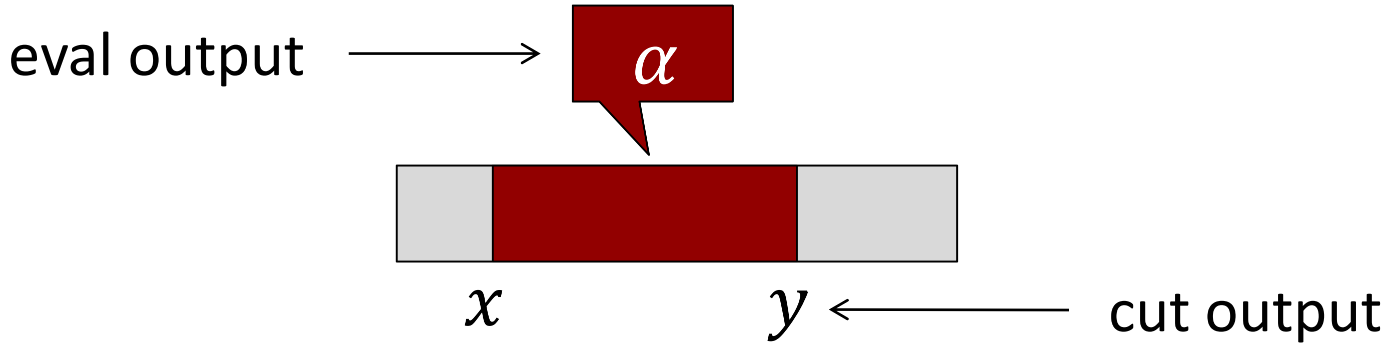
- This is EF and therefore proportional.
  - Why?

Input Model

- How do we measure the "time complexity" of a cake-cutting algorithm for n players?
- Typically, time complexity is a function of the length of input encoded as binary.
- Our input consists of functions V_i, which requires infinite bits to encode.
- We want running time just as a function of n.

Robertson-Webb Model

- We restrict access to valuations V_i's through two types of queries:
  - $\text{Eval}_i(x, y)$ returns $V_i([x, y])$
  - $\text{Cut}_i(x, \alpha)$ returns y such that $V_i([x, y]) = \alpha$



Robertson-Webb Model

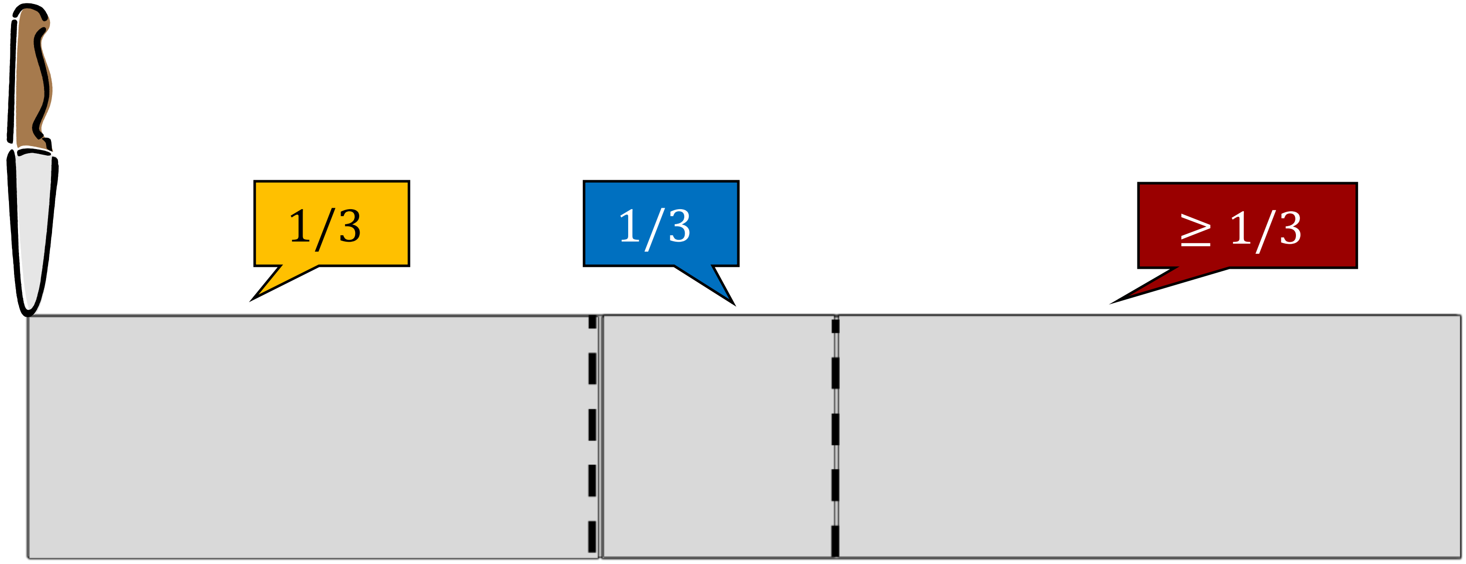
- Two types of queries:
  - $\text{Eval}_i(x, y) = V_i([x, y])$
  - $\text{Cut}_i(x, \alpha) = y$ s.t. $V_i([x, y]) = \alpha$
- Question: How many queries are needed to find an EF allocation when $n = 2$?
- Answer: 2
  - Why?

Dubins-Spanier

- Protocol for finding a proportional allocation for n players

- Referee starts at 0, and continuously moves knife to the right.
- Repeat: when the piece to the left of knife is worth $1/n$ to a player, the player shouts "stop", gets the piece, and exits.
- The last player gets the remaining piece.

DUBINS-SPANIER



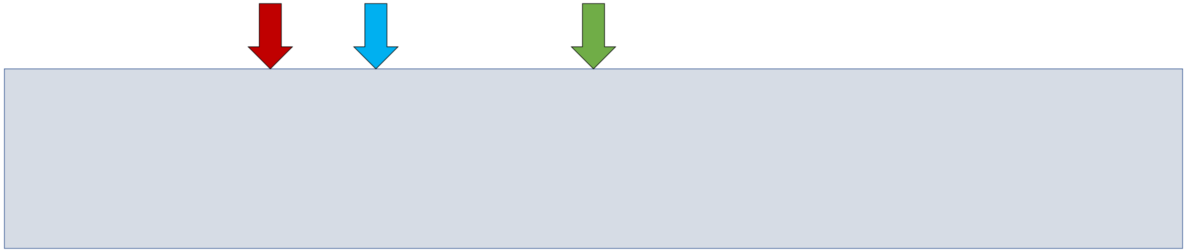
Dubins-Spanier

- Moving knife is not really needed.

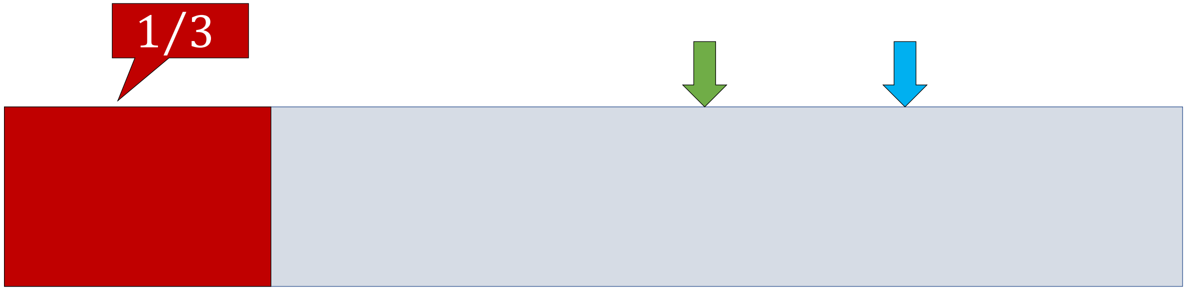
- At each stage, we can ask each remaining player a cut query to mark his $1/n$ point in the remaining cake.

- Move the knife to the leftmost mark.

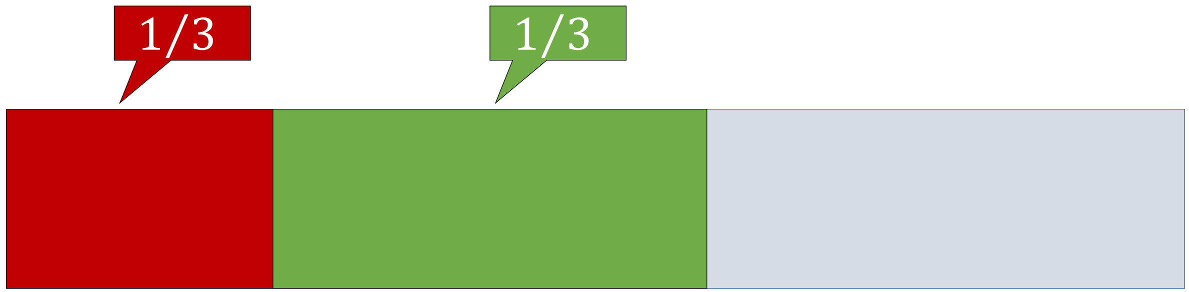
Dubins-Spanier



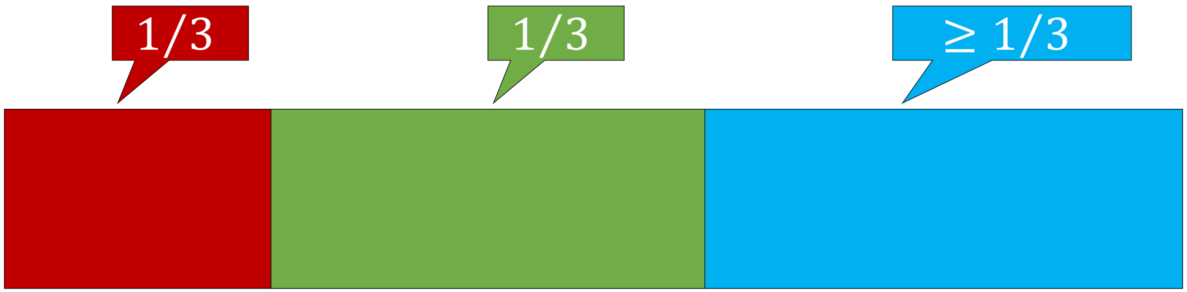
Dubins-Spanier



Dubins-Spanier



DUBINS-SPANIER



Dubins-Spanier

- Question: What is the complexity of the Dubins-Spanier protocol in the Robertson-Webb model?

1. $\Theta(n)$
2. $\Theta(n \log n)$
3. $\Theta(n^2)$
4. $\Theta(n^2 \log n)$

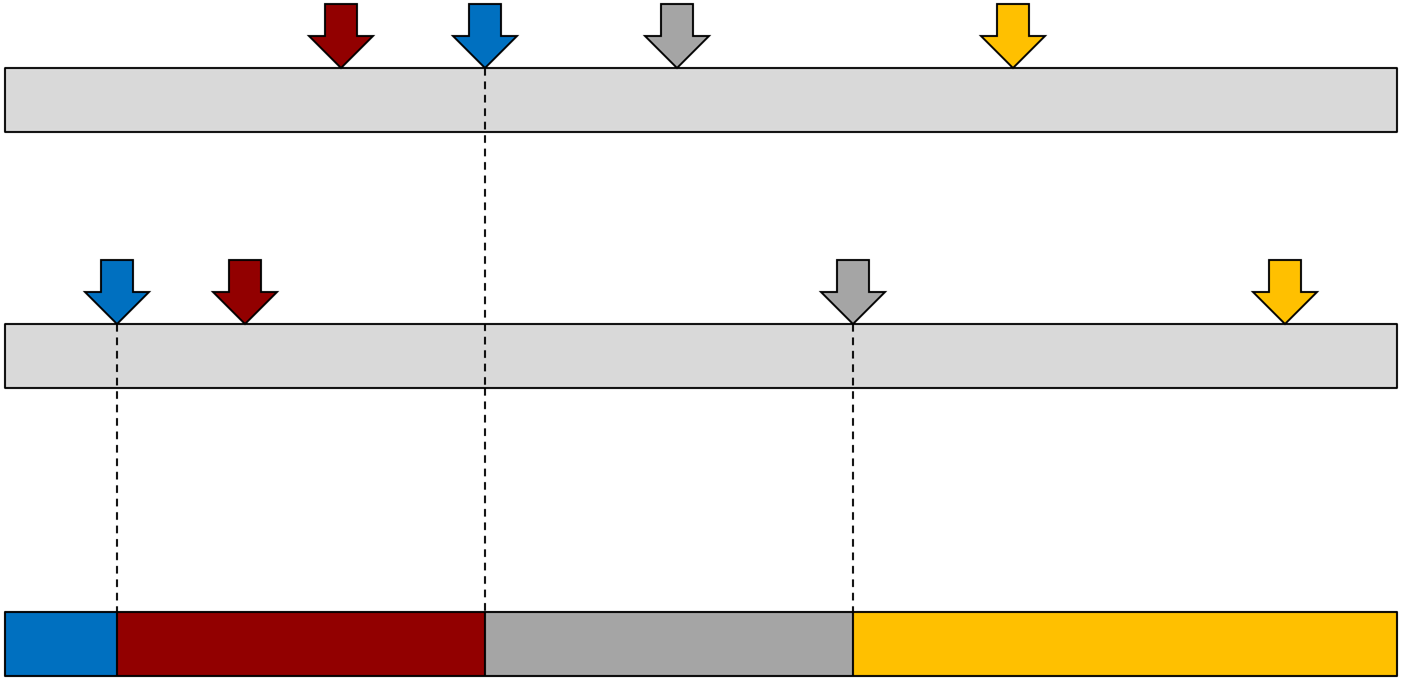
EVEN-PAZ

- Input: Interval $[x, y]$, number of players n
  - Assume $n = 2^k$ for some k
- If $n = 1$, give $[x, y]$ to the single player.
- Otherwise, let each player i mark z_i s.t.

$$V_i([x, z_i]) = \frac{1}{2} V_i([x, y])$$

- Let z^* be the $n/2$ mark from the left.
- Recurse on $[x, z^*]$ with the left $n/2$ players, and on $[z^*, y]$ with the right $n/2$ players.

EVEN-PAZ



EVEN-PAZ

- Theorem: EVEN-PAZ returns a Prop allocation.
- Proof:
  - Inductive proof. We want to prove that if player i is allocated piece A_i when $[x, y]$ is divided between n players, $V_i(A_i) \geq (1/n)V_i([x, y])$
    - Then Prop follows because initially $V_i([x, y]) = V_i([0,1]) = 1$
  - Base case: $n = 1$ is trivial.
  - Suppose it holds for $n = 2^{k-1}$. We prove for $n = 2^k$.
  - Take the 2^{k-1} left players.
    - Every left player i has $V_i([x, z^*]) \geq (1/2)\, V_i([x, y])$
    - If it gets A_i, by induction, $V_i(A_i) \geq \frac{1}{2^{k-1}}\, V_i([x, z^*]) \geq \frac{1}{2^k}\, V_i([x, y])$

Even-Paz

- Question: What is the complexity of the Even-Paz protocol in the Robertson-Webb model?

1. $\Theta(n)$
2. $\Theta(n \log n)$
3. $\Theta(n^2)$
4. $\Theta(n^2 \log n)$

Complexity of Proportionality

- Theorem [Edmonds and Pruhs, 2006]: Any proportional protocol needs $\Omega(n \log n)$ operations in the Robertson-Webb model.

- Thus, the EVEN-PAZ protocol is (asymptotically) provably optimal!

Envy-Freeness?

- "I suppose you are also going to give such cute algorithms for finding envy-free allocations?"

- Bad luck. For n-player EF cake-cutting:
  - [Brams and Taylor, 1995] give an unbounded EF protocol.
  - [Procaccia 2009] shows $\Omega(n^2)$ lower bound for EF.
  - Last year, the long-standing major open question of "bounded EF protocol" was resolved!
  - [Aziz and Mackenzie, 2016]: $O(n^{n^{n^{n^{n^n}}}})$ protocol!
    - Not a typo!

Other Desiderata

- There are two more properties that we often desire from an allocation.

- Pareto optimality (PO)
  - Notion of efficiency
  - Informally, it says that there should be no "obviously better" allocation

- Strategyproofness (SP)
  - No player should be able to gain by misreporting her valuation

Strategyproofness (SP)

- For deterministic mechanisms
  - "Strategyproof": No player should be able to increase her *utility* by misreporting her valuation, irrespective of what other players report.

- For randomized mechanisms
  - "Strategyproof-in-expectation": No player should be able to increase her *expected utility* by misreporting.
  - For simplicity, we'll call this strategyproofness, and assume we mean "in expectation" if the mechanism is randomized.

Strategyproofness (SP)

- Deterministic
  - Bad news!
  - Theorem [Menon & Larson '17] : No deterministic SP mechanism is (even approximately) proportional.

- Randomized
  - Good news!
  - Theorem [Chen et al. '13, Mossel & Tamuz '10]: There is a randomized SP mechanism that always returns an envy-free allocation.

Perfect Partition

- Theorem [Lyapunov '40]:
  - There always exists a "perfect partition" $(B_1, \ldots, B_n)$ of the cake such that $V_i(B_j) = {}^1/_n$ for every $i, j \in [n]$.
  - Every agent values every bundle equally.

- Theorem [Alon '87]:
  - There exists a perfect partition that only cuts the cake at $poly(n)$ points.
  - In contrast, Lyapunov's proof is non-constructive, and might need an unbounded number of cuts.

Perfect Partition

- Q: Can you use an algorithm for computing a perfect partition as a black-box to design a randomized SP-in-expectation+EF mechanism?
  - Yes! Compute a perfect partition, and assign the n bundles to the n players uniformly at random.
  - Why is this EF?
    - Every agent values every bundle at $1/n$.
  - Why is this SP-in-expectation?
    - Because an agent is assigned a random bundle, her expected utility is $1/n$, irrespective of what she reports.

Pareto Optimality (PO)

- Definition
  - We say that an allocation $A = (A_1, \dots, A_n)$ is PO if there is no alternative allocation $B = (B_1, \dots, B_n)$ such that

  1. Every agent is at least as happy: $V_i(B_i) \geq V_i(A_i), \forall i \in N$
  2. Some agent is strictly happier: $V_i(B_i) > V_i(A_i), \exists i \in N$

  - I.e., an allocation is PO if there is no “better” allocation.

- Q: Is it PO to give the entire cake to player 1?
- A: Not necessarily. But yes if player 1 values “every part of the cake positively”.

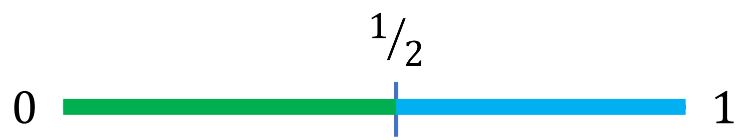
PO + EF

- Theorem [Weller '85]:
  - There always exists an allocation of the cake that is both envy-free and Pareto optimal.

- One way to achieve PO+EF:
  - Nash-optimal allocation: $\text{argmax}_A \prod_{i \in N} V_i(A_i)$
  - Obviously, this is PO. The fact that it is EF is non-trivial.
  - This is named after John Nash.
    - Nash social welfare = product of utilities
    - Different from utilitarian social welfare = sum of utilities

Nash-Optimal Allocation

- Example:
  - Green player has value 1 distributed over $[0, {}^2/_3]$
  - Blue player has value 1 distributed over $[0,1]$
  - Without loss of generality (why?) suppose:
    - Green player gets x fraction of $[0, {}^2/_3]$
    - Blue player gets the remaining $1 - x$ fraction of $[0, {}^2/_3]$ AND all of $[{}^2/_3, 1]$.
  - Green's utility = x, blue's utility = $(1 - \mathrm{x}) \cdot \frac{2}{3} + \frac{1}{3} = \frac{3-2x}{3}$
  - Maximize: $x \cdot \frac{3-2x}{3} \Rightarrow x = {}^3/_4$ (${}^3/_4$ fraction of ${}^2/_3$ is ${}^1/_2$).

Allocation 

Green has utility $\frac{3}{4}$

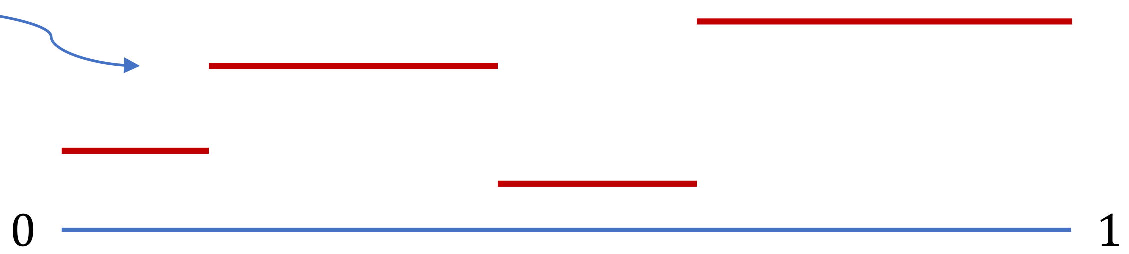
Blue has utility $\frac{1}{2}$

Problem with Nash Solution

- Difficult to compute in general
  - I believe it should require an unbounded number of queries in the Robertson-Webb model. But I can't find such a result in the literature.

- Theorem [Aziz & Ye '14]:
  - For *piecewise constant* valuations, the Nash-optimal solution can be computed in polynomial time.

The density function of a piecewise constant valuation looks like this



Interlude: Homogeneous Divisible Goods

- Suppose there are m homogeneous divisible goods
  - Each good can be divided fractionally between the agents
- Let $x_{i,g}$ = fraction of good g that agent i gets
  - Homogeneous = agent doesn't care which "part"
    - E.g., CPU or RAM
- Special case of cake-cutting
  - Line up the goods on $[0,1]$ → piecewise uniform valuations

Interlude: Homogeneous Divisible Goods

- Nash-optimal solution:

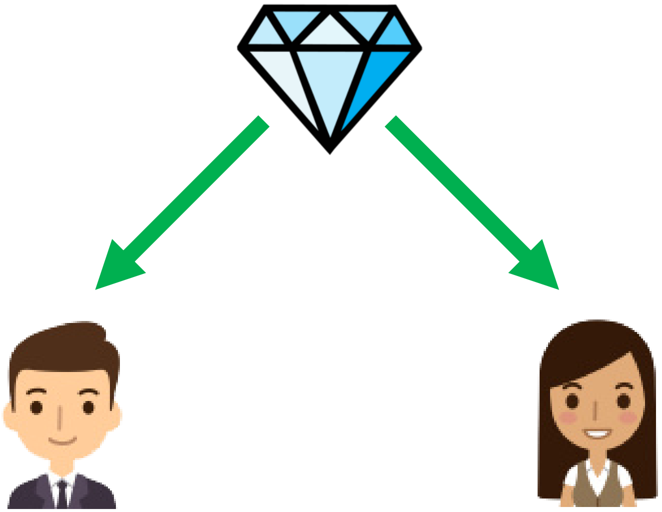
  Maximize $\sum_i \log U_i$

  $$
  \begin{array}{ll}
  U_i = \Sigma_g \, x_{i,g} * v_{i,g} & \forall i \\
  \Sigma_i \, x_{i,g} = 1 & \forall g \\
  x_{i,g} \in [0,1] & \forall i, g
  \end{array}
  $$

- Gale-Eisenberg Convex Program
  - Polynomial time solvable

Indivisible Goods

- Goods which cannot be shared among players
  - E.g., house, painting, car, jewelry, …
- Problem: Envy-free allocations may not exist!



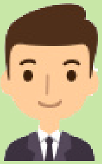
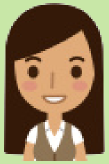
Indivisible Goods: Setting

				
	8	7	20	5
	9	11	12	8
	9	10	18	3

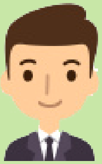
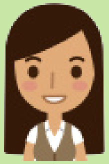
Given such a matrix of numbers, assign each good to a player.

We assume additive values. So, e.g., V$(\{$, $\}) = 8 + 7 = 15$

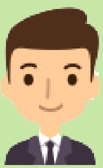
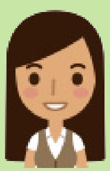
Indivisible Goods

				
	8	7	20	5
	9	11	12	8
	9	10	18	3

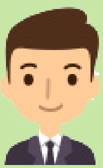
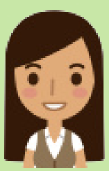
Indivisible Goods

				
	8	7	20	5
	9	11	12	8
	9	10	18	3

Indivisible Goods

				
	8	7	20	5
	9	11	12	8
	9	10	18	3

Indivisible Goods

				
	8	7	20	5
	9	11	12	8
	9	10	18	3

Indivisible Goods

- Envy-freeness up to one good (EF1):

$$\forall i, j \in N, \exists g \in A_j : V_i(A_i) \geq V_i\left(A_j \backslash \{g\}\right)$$

  - Technically, we need either this or $A_j = \emptyset$.
  - "If i envies j, there must be some good in j's bundle such that removing it would make i envy-free of j."

- Does there always exist an EF1 allocation?

EF1

- Yes! We can use Round Robin.
  - Agents take turns in cyclic order: $1,2, \dots, n, 1,2, \dots, n, \dots$
  - In her turn, an agent picks the good she likes the most among the goods still not picked by anyone.

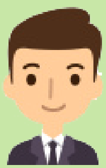
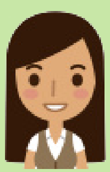
- Observation: This always yields an EF1 allocation.
  - Informal proof on the board.

- Sadly, on some instances, this returns an allocation that is not Pareto optimal.

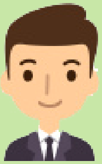
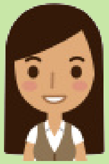
EF1+PO?

- Nash welfare to rescue!
- Theorem [Caragiannis et al. '16]:
  - The allocation $\text{argmax}_A \prod_{i \in N} V_i(A_i)$ is EF1 + PO.
  - Note: This maximization is over only "integral" allocations that assign each good to some player in whole.
  - Note: Subtle tie-breaking if all allocations have zero Nash welfare.
    - Step 1: Choose a subset of players $S \subseteq N$ with largest $|S|$ such that it is possible to give a positive utility to every player in S simultaneously.
    - Step 2: Choose $\text{argmax}_A \prod_{i \in S} V_i(A_i)$

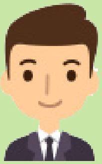
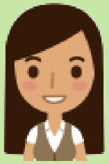
Integral Nash Allocation

				
	8	7	20	5
	9	11	12	8
	9	10	18	3

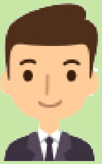
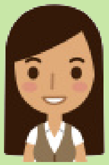
20 * 8 * (9+10) = 3040

				
	8	7	20	5
	9	11	12	8
	9	10	18	3

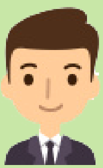
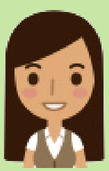
$(8+7) * 8 * 18 = 2160$

				
	8	7	20	5
	9	11	12	8
	9	10	18	3

$$8 * (12+8) * 10 = 1600$$

				
	8	7	20	5
	9	11	12	8
	9	10	18	3

20 * (11+8) * 9 = 3420

				
	8	7	20	5
	9	11	12	8
	9	10	18	3

Computation

- For indivisible goods, Nash-optimal solution is strongly NP-hard to compute
  - That is, remains NP-hard even if all values in the matrix are bounded

- Open Question: If our goal is EF1+PO, is there a different polynomial time algorithm?
  - Not sure. But a recent paper gives a pseudo-polynomial time algorithm for EF1+PO
    - Time is polynomial in n, m, and $\max_{i,g} V_i(\{g\})$.

Other Fairness Notions

- Maximin Share Guarantee (MMS):
  - Generalization of "cut and choose" for n players
  - MMS value of player i =
    - The highest value player i can get…
    - If *she* divides the goods into n bundles…
    - But receives the worst bundle for her ("worst case guarantee")
  - Let $\mathcal{P}_n(M)$ denote the family of partitions of the set of goods M into n bundles.

$$MMS_i = \max_{(B_1,\dots,B_n)\in\mathcal{P}_n(M)} \min_{k\in\{1,\dots,n\}} V_i(B_k).$$

  - An allocation is α-MMS if every player i receives value at least $\alpha * MMS_i$.

Other Fairness Notions

- Maximin Share Guarantee (MMS)
  - [Procaccia, Wang '14]:
    There is an example in which no MMS allocation exists.
  - [Procaccia, Wang '14]:
    A $^2/_3$ - MMS allocation always exists.
  - [Ghodsi et al. '17]:
    A $^3/_4$ - MMS allocation always exists.
  - [Caragiannis et al. '16]:
    The Nash-optimal solution is $\frac{2}{1+\sqrt{4n-3}}$ −MMS, and this is the best possible guarantee.

Stronger Fairness

- Open Question: Does there always exist an EFx allocation?
- EF1: $\forall i, j \in N, \exists g \in A_j : \; V_i(A_i) \geq V_i\left(A_j \setminus \{g\}\right)$
  - Intuitively, i doesn't envy j if she gets to remove her most valued item from j's bundle.
- EFx: $\forall i, j \in N, \forall g \in A_j : \; V_i(A_i) \geq V_i\left(A_j \setminus \{g\}\right)$
  - Note: Need to quantify over g such that $V_i(\{g\}) > 0$.
  - Intuitively, i doesn't envy j even if she removes her least positively valued item from j's bundle.

Stronger Fairness

- The difference between EF1 and EFx:
  - Suppose there are two players and three goods with values as follows.

	A	B	C
P1	5	1	10
P2	0	1	10

  - If you give {A} → P1 and {B,C} → P2, it's EF1 but not EFx.
    - EF1 because if P1 removes C from P2's bundle, all is fine.
    - Not EFx because removing B doesn't eliminate envy.
  - Instead, {A,B} → P1 and {C} → P2 would be EFx.

Allocation of Bads

- Negative utilities (costs instead of values)
  - Let $c_{i,b}$ be the cost of player i for bad b.
    - $C_i(S) = \sum_{b \in S} c_{i,b}$
  - EF: $\forall i, j \;\; C_i(A_i) \le C_i(A_j)$
  - PO: There should be no alternative allocation in which no player has more cost, and some player has less cost.

- Divisible bads
  - EF + PO allocation always exists, like for divisible goods.
    - One way to achieve is through "Competitive Equilibria" (CE).
    - For divisible goods, Nash-optimal allocation is the unique CE.
    - For bads, there are exponentially many CE.

Allocation of Bads

- Indivisible bads
  - EF1: $\forall i,j \;\; \exists b \in A_i \;\; c_i(A_i \setminus \{b\}) \le c_i\left(A_j\right)$
  - EFx: $\forall i,j \;\; \forall b \in A_i \; c_i(A_i \setminus \{b\}) \le c_i\left(A_j\right)$
    - Note: Again, we need to restrict to b such that $c_{i,b} > 0$
  - Open Question 1:
    - Does an EF1 + PO allocation always exist?
  - Open Question 2:
    - Does an EFx allocation always exist?
  - More open questions related to relaxations of proportionality