Fair and Efficient Social Decision-Making

CSCI 699

Fair Division: Divisible Goods

Evi Micha

Credit for the slides : Nisarg Shah and Ariel Procaccia

Cake-Cutting

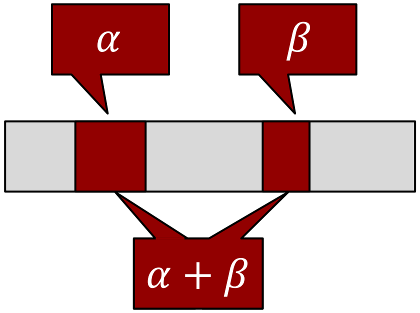
Cake-Cutting

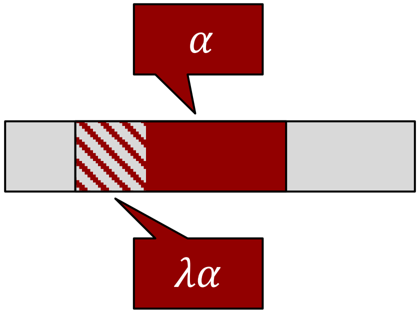
- A heterogeneous divisible good
  - Heterogeneous = same part may be valued differently by different agents
  - Divisible = can be divided between agents
- Cake $C = [0,1]$
  - Almost without loss of generality
- Agents $N = \{1, \dots, n\}$
- Piece of cake $X \subseteq [0,1]$ = finite union of disjoint intervals
- Allocation $A = (A_1, \dots, A_n)$
  - Partition of the cake where each A_i is a piece of the cake

Agent Valuations

- Valuation of agent i is given by an integrable value density function $f_i: [0,1] \to \mathbb{R}_+$
  - Her value for a piece of cake X is $V_i(X) = \int_{x \in X} f_i(x) dx$

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



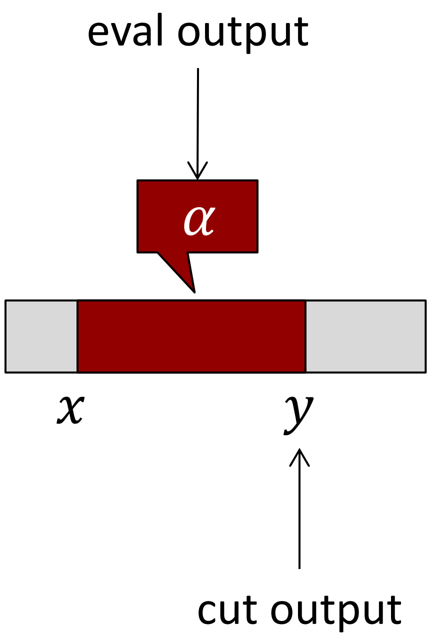


- WLOG
  - Normalized: $V_i([0,1]) = 1$

Complexity

- Inputs are functions
  - Infinitely many bits may be needed to fully represent the input
  - Query complexity is more useful

- Robertson-Webb Model
  - $\text{Eval}_i(x, y)$ returns $v_i([x, y])$
  - $\text{Cut}_i(x, \alpha)$ returns y such that $v_i([x, y]) = \alpha$



Fairness Goals

- What kind of fairness might we want from an allocation A?
- Proportionality (Prop):

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

- Envy-Freeness (EF):

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

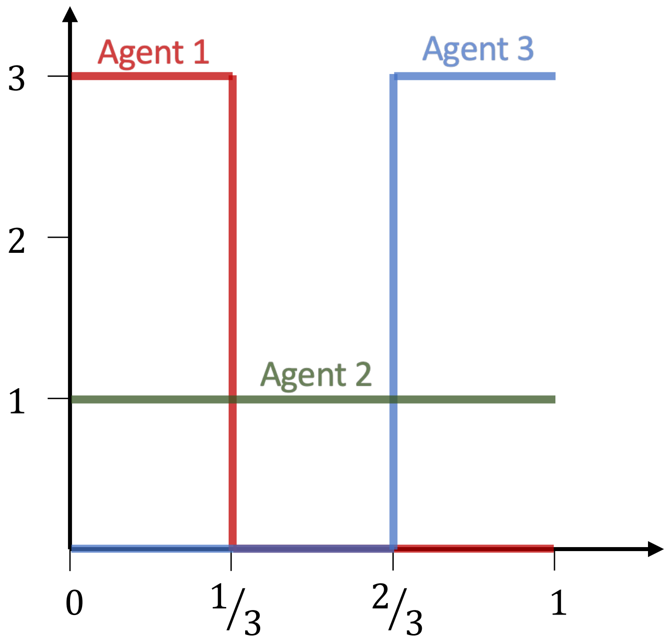
- Equitability (EQ):

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

Only makes sense with normalization

Example

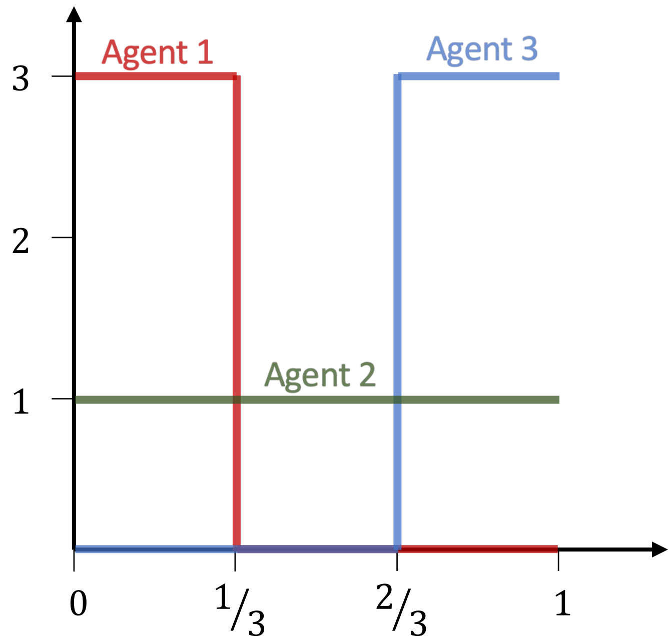
- Value density functions



- Agent 1 wants $[0, ^1/_3]$ uniformly and does not want anything else
- Agent 2 wants the entire cake uniformly
- Agent 3 wants $[^2/_3, 1]$ uniformly and does not want anything else

Example

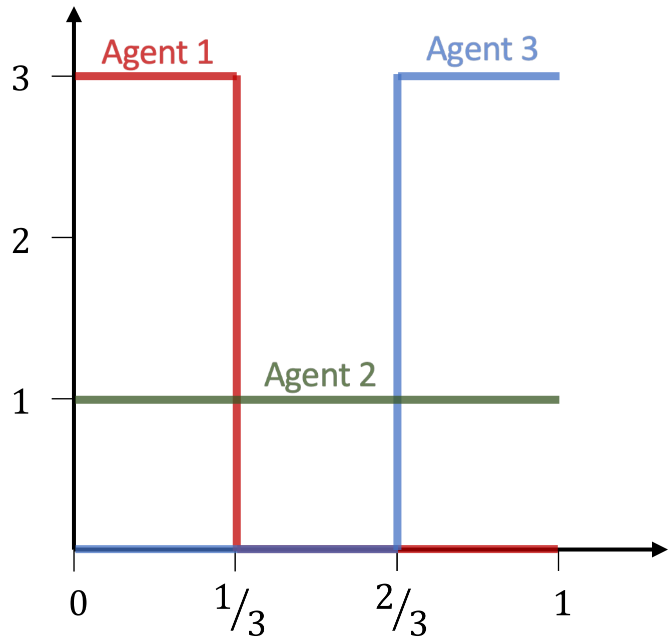
- Value density functions



- Consider the following allocation
- $A_1 = [0, {}^1/_9] \Rightarrow v_1(A_1) = {}^1/_3$
- $A_2 = [{}^1/_9, {}^8/_9] \Rightarrow v_2(A_2) = {}^7/_9$
- $A_3 = [{}^8/_9, 1] \Rightarrow v_3(A_3) = {}^1/_3$
- The allocation is proportional, but not envy-free or equitable

Example

- Value density functions



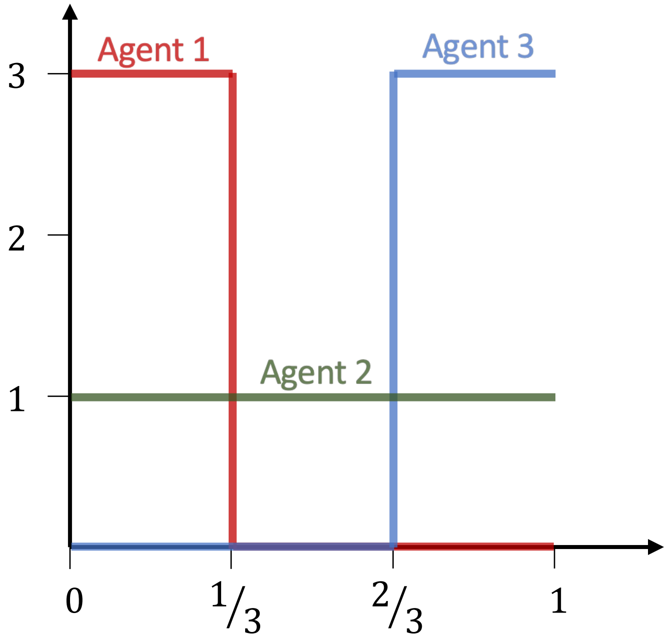
- Consider the following allocation

- $A_1 = [0, {}^1\!/_6] \Rightarrow v_1(A_1) = {}^1\!/_2$
- $A_2 = [{}^1\!/_6, {}^5\!/_6] \Rightarrow v_2(A_2) = {}^2\!/_3$
- $A_3 = [{}^5\!/_6, 1] \Rightarrow v_3(A_3) = {}^1\!/_2$

- The allocation is proportional and envy-free, but not equitable

Example

- Value density functions



- Consider the following allocation

- $A_1 = [0, {}^1/_5] \Rightarrow v_1(A_1) = {}^3/_5$
- $A_2 = [{}^1/_5, {}^4/_5] \Rightarrow v_2(A_2) = {}^3/_5$
- $A_3 = [{}^4/_5, 1] \Rightarrow v_3(A_3) = {}^3/_5$

- The allocation is proportional, envy-free, and equitable

Relations Between Fairness Desiderata

- 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

Relations Between Fairness Desiderata

- Envy-freeness implies proportionality
  - Summing $v_i(A_i) \geq v_i(A_j)$ over all j gives proportionality
- For 2 agents, proportionality also implies envy-freeness
  - Hence, they are equivalent.
- Equitability is incomparable to proportionality and envy-freeness
  - E.g. if each agent has value 0 for her own allocation and 1 for the other agent’s allocation, it is equitable but not proportional or envy-free.

Existence

- Theorem [Alon, 1987]
  Suppose the value density function f_i of each agent valuation v_i is continuous. Then, we can cut the cake at $n^2 - n$ places and rearrange the $n^2 - n + 1$ intervals into n pieces $A_1, \dots, A_n$ such that

$$v_i(A_j) = {}^1/_n, \forall i, j \in N$$

- This is called a "perfect partition"
  - It is trivially envy-free (thus proportional) and equitable
- As we will later see, this cannot be found with finitely many queries in Robertson-Webb model

Proportionality

Cut-and-Choose

- Algorithm for $n = 2$ agents

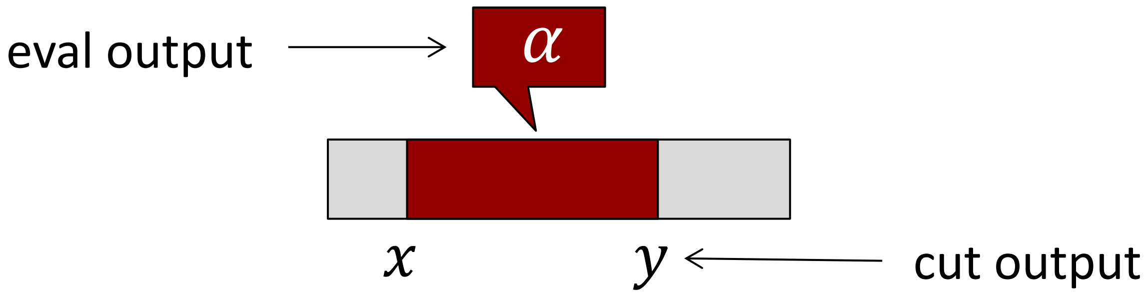
- Agent 1 divides the cake into two pieces X, Y s.t.
$$V_1(X) = V_1(Y) = 1/2$$

- Agent 2 chooses the piece she prefers.

- This is EF and therefore proportional.
  - Why?

Robertson-Webb Model

- Two types of queries to an agent's valuation function V_i
  - $\text{Eval}_i(x, y)$ returns $V_i([x, y])$
  - $\text{Cut}_i(x, \alpha)$ returns the smallest y such that $V_i([x, y]) = \alpha$
    - If no such y exists, then it returns 1



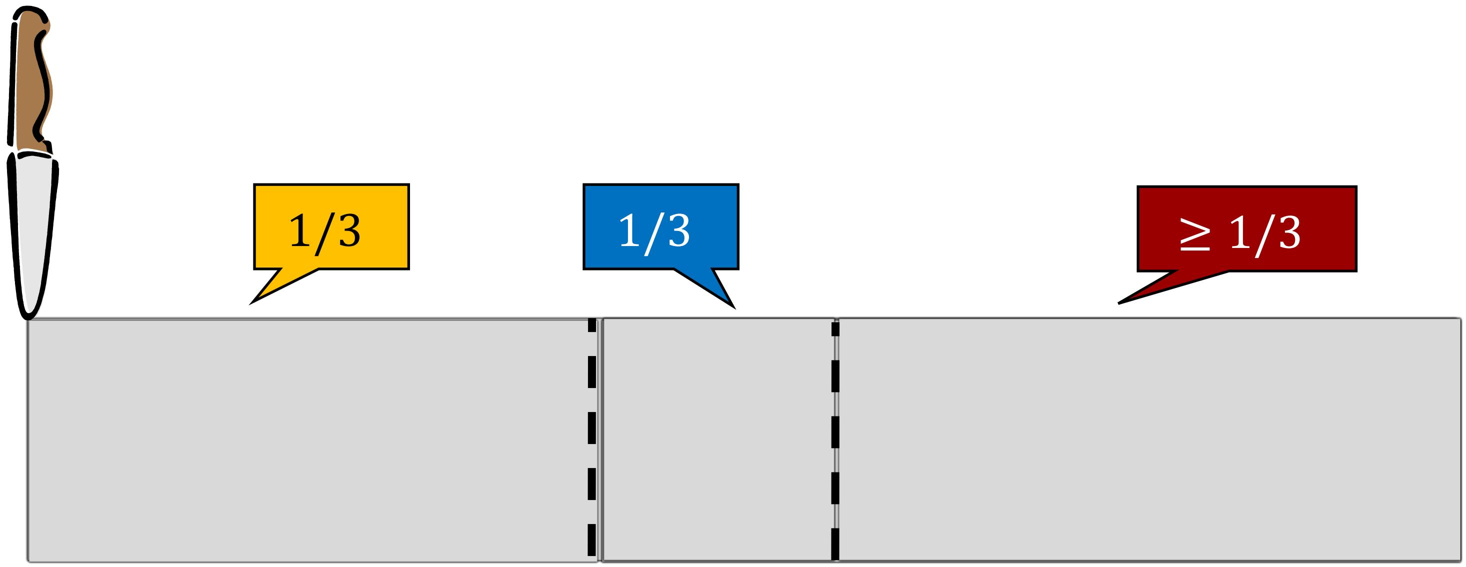
- Question:
  - How many queries are needed to find an EF allocation when $n = 2$?

Dubins-Spanier

- Protocol for finding a proportional allocation for n agents

- Referee starts with a knife at 0
- Referee continuously moves the knife to the right
- Repeat $n - 1$ times: Whenever the piece to the left of knife is worth $1/n$ to a agent, the agent shouts “stop”, gets the piece, and exits.
- The last agent gets the remaining piece.

Dubins-Spanier



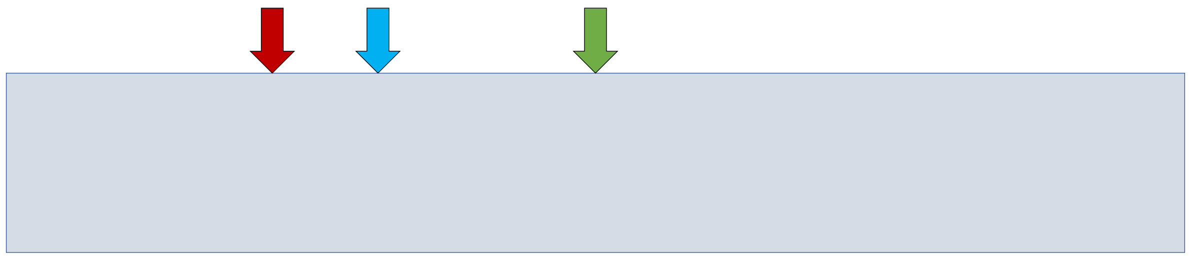
Dubins-Spanier

- Moving a knife continuously is not really needed.

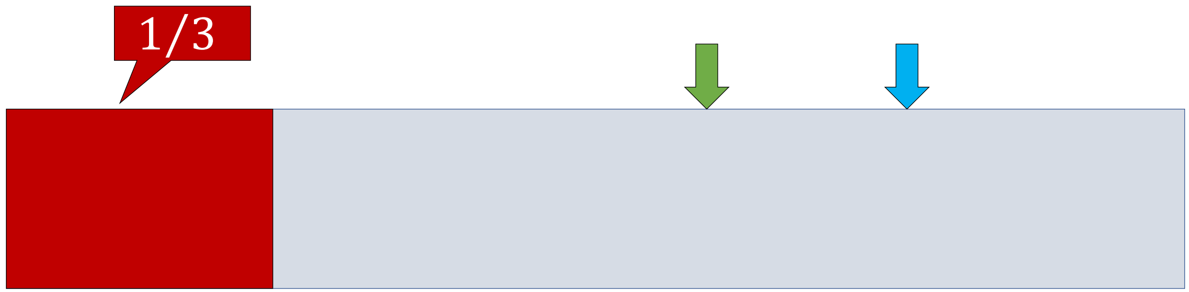
- At each stage, we can ask each remaining agent a cut query to mark his $1/n$ point in the remaining cake.

- Move the knife to the leftmost mark.

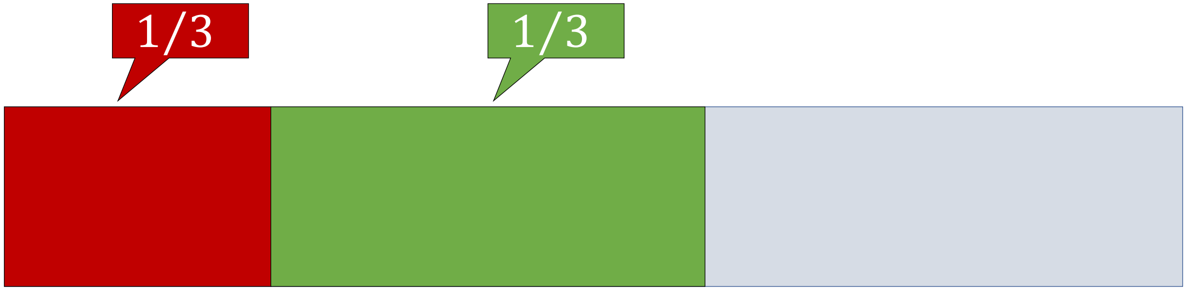
Dubins-Spanier



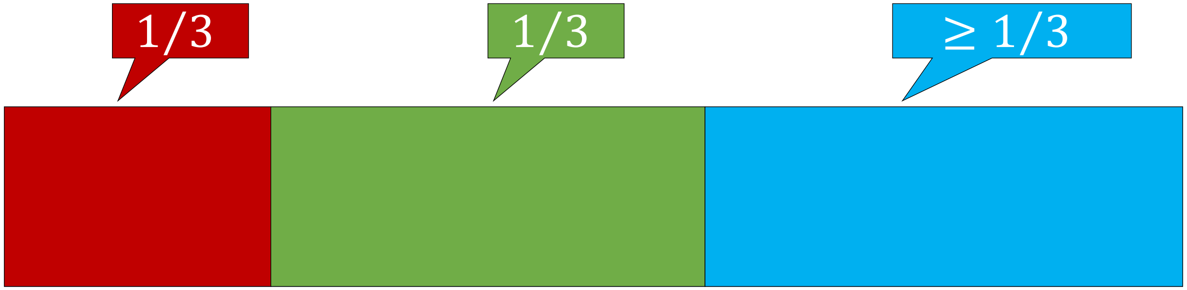
Dubins-Spanier



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

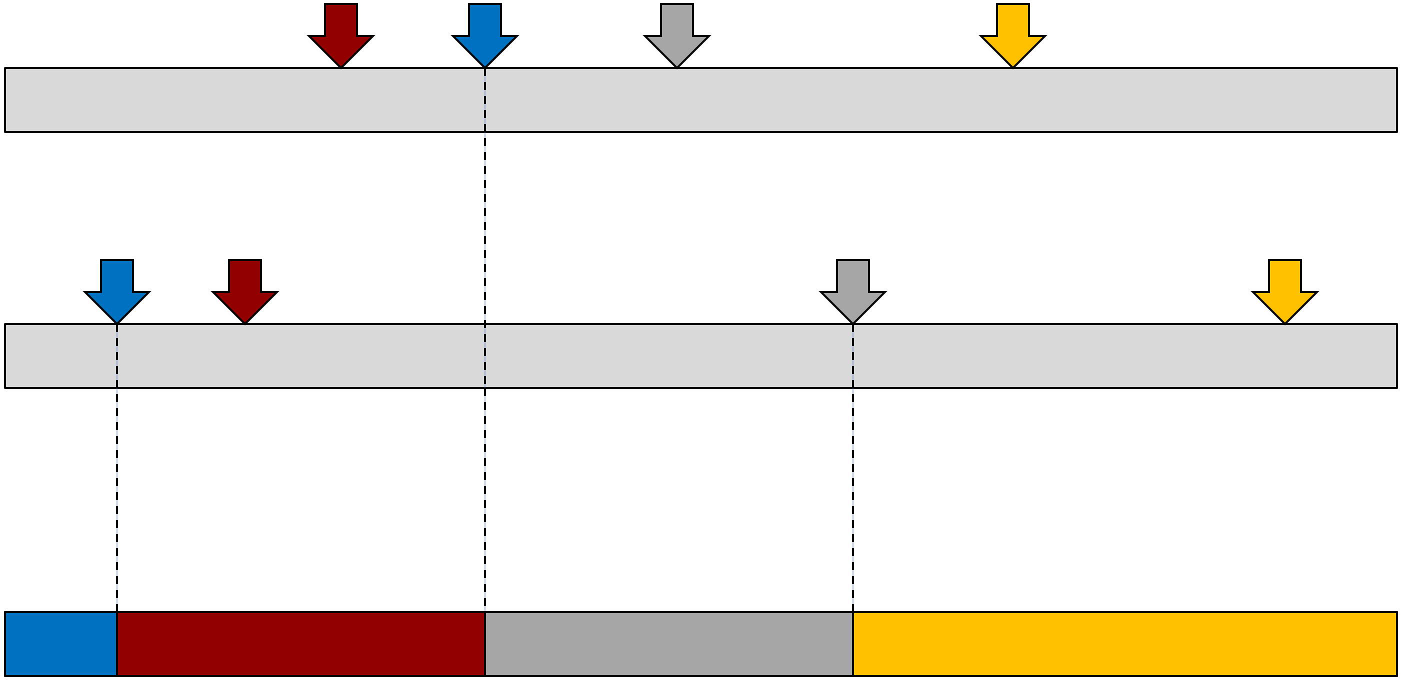
- Input: Interval $[x, y]$, number of agents n
  - Assume $n = 2^k$ for some k

- If $n = 1$, give $[x, y]$ to the single agent.
- Otherwise, let each agent i mark z_i s.t.

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

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

Even-Paz



EVEN-PAZ

- Theorem: EVEN-PAZ returns a Prop allocation.
- Proof:
  - Inductive proof. We want to prove that if agent i is allocated piece A_i when $[x, y]$ is divided between n agents, $V_i(A_i) \geq (1/n)V_i([x, y])$
    - 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 agents.
    - Every left agent 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

Envy-Freeness : Few Agents

- $n = 2$ agents : CUT-AND-CHOOSE (2 queries)
- $n = 3$ agents : SELFRIDGE-CONWAY (14 queries)

Gets complex pretty quickly!

Suppose we have three players **P1**, **P2** and **P3**. Where the procedure gives a criterion for a decision it means that criterion gives an optimum choice for the player.

1. **P1** divides the cake into three pieces he considers of equal size.
2. Let's call **A** the largest piece according to **P2**.
3. **P2** cuts off a bit of **A** to make it the same size as the second largest. Now **A** is divided into: the trimmed piece **A1** and the trimmings **A2**. Leave the trimmings **A2** to the side for now.
   - If **P2** thinks that the two largest parts are equal (such that no trimming is needed), then each player chooses a part in this order: **P3**, **P2** and finally **P1**.
4. **P3** chooses a piece among **A1** and the two other pieces.
5. **P2** chooses a piece with the limitation that if **P3** didn't choose **A1**, **P2** must choose it.
6. **P1** chooses the last piece leaving just the trimmings **A2** to be divided.

It remains to divide the trimmings **A2**. The trimmed piece **A1** has been chosen by either **P2** or **P3**; let's call the player who chose it **PA** and the other player **PB**.

1. **PB** cuts **A2** into three equal pieces.
2. **PA** chooses a piece of **A2** - we name it **A21**.
3. **P1** chooses a piece of **A2** - we name it **A22**.
4. **PB** chooses the last remaining piece of **A2** - we name it **A23**.

Envy-Freeness : Few Agents

- [Brams and Taylor, 1995]
  - The first finite (but unbounded) protocol for any number of agents

- [Aziz and Mackenzie, 2016a]
  - The first bounded protocol for 4 agents (at most 203 queries)

- [Amanatidis et al., 2018]
  - A simplified version of the above protocol for 4 agents (at most 171 queries)

Envy-Freeness

- Theorem [Aziz and Mackenzie, 2016b]
  - There exists a bounded protocol for computing an envy-free allocation with n agents, which requires $O(n^{n^{n^{n^{n^n}}}})$ queries
  - After $O(n^{2n+3})$ queries, the protocol can output a partial allocation that is both proportional and envy-free

- What about lower bounds?

Complexity of Envy-Freeness

- Theorem [Procaccia, 2009]
  Any protocol for finding an envy-free allocation requires $\Omega(n^2)$ queries.

> **Open Problem**
>
> Bridge the gap between $O(n^{n^{n^{n^{n^n}}}})$ upper bound and $\Omega(n^2)$ lower bound for envy-free cake-cutting

- Theorem [Stromquist, 2008]
  There is no finite (even unbounded) protocol for finding a simple envy-free allocation for $n \geq 3$ agents.

Equitability

Upper Bound: $n = 2$ Agents

- Existence
  - Suppose we cut the cake at x to form pieces $[0, x]$ and $[x, 1]$
  - Let $f(x) = v_1([0, x]) - v_2([x, 1])$
    - Note that $f(0) = -1$, $f(1) = 1$, and f is continuous
  - By the intermediate value theorem: $\exists x^*$ such that $f(x^*) = 0$
  - Allocation $A_1 = [0, x^*]$ and $A_2 = [x^*, 1]$ is equitable

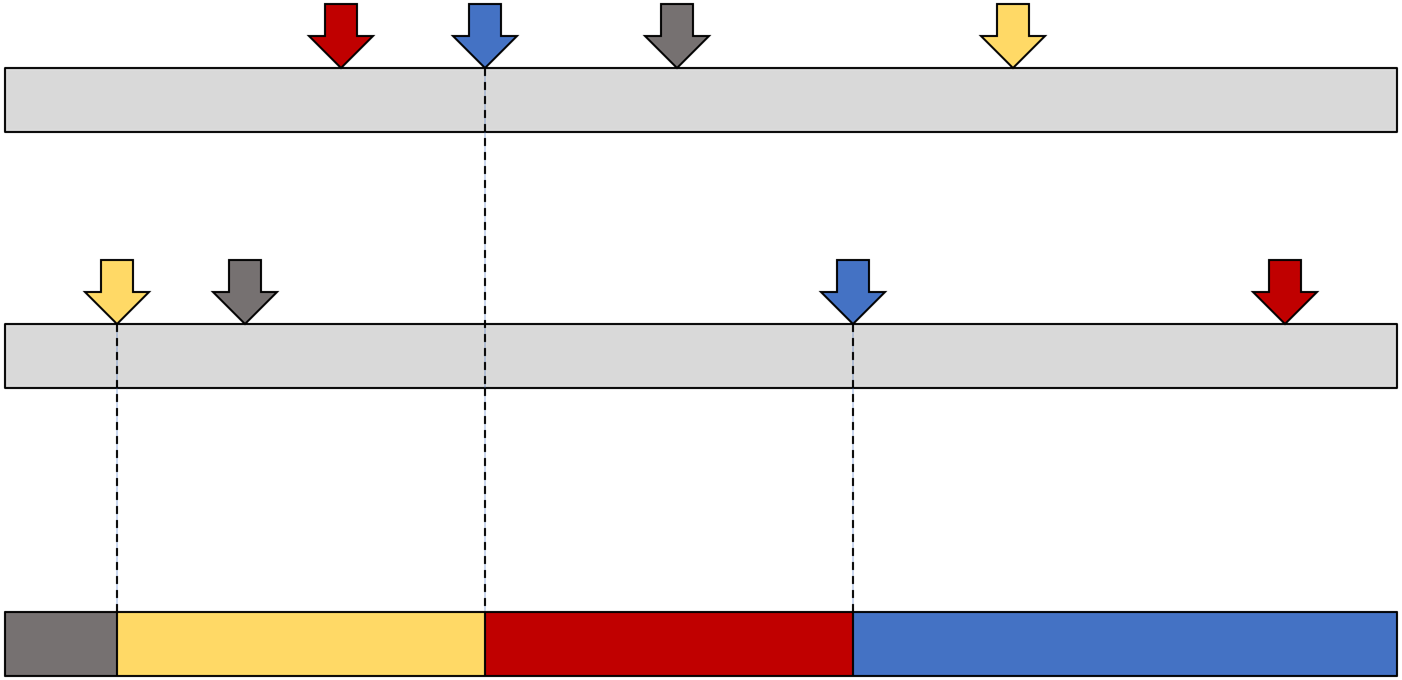
- Theorem [Cechlárová and Pillárová, 2012]
  - Using binary search for x^*, we can find an ϵ-equitable allocation for 2 agents with $O(\ln(1/\epsilon))$ queries.

Upper Bound: $n > 2$ Agents

- Theorem [Cechlárová and Pillárová, 2012]
  - This technique can be extended to n agents to find an ϵ-equitable allocation in $O(n \ln(1/\epsilon))$ queries.

- Theorem [Procaccia and Wang, 2017]
  - There exists a protocol for n agents which finds an ϵ-equitable allocation in $O(1/\epsilon \ \ln(1/\epsilon))$ queries.
  - Intuition:
    - If $n \leq 1/\epsilon$, use above protocol for finding an equitable ϵ-equitable allocation.
    - If $n > 1/\epsilon$, use a variant of the Evan-Paz algorithm to find an *anti-proportional* allocation where $n' = \lceil 1/\epsilon \rceil$ agents get value *at most* $1/n'$, and the rest receive nothing.
      - While this is a “bad” allocation, it is ϵ-equitable.

Antiproportinal Even-Paz



Lower Bound

- Theorem [Procaccia and Wang, 2017]
  Any protocol for finding an ϵ-equitable allocation must require $\Omega\left(\frac{\ln(1/\epsilon)}{\ln\ln(1/\epsilon)}\right)$ queries.

- Theorem [Procaccia and Wang, 2017]
  There is no finite (even if unbounded) protocol for finding an equitable allocation.

Efficiency

Efficiency

- Weak Pareto optimality (WPO)
  - Allocation A is weakly Pareto optimal if there is no allocation B such that $v_i(B_i) > v_i(A_i)$ for all $i \in N$.
  - "Can't make everyone happier"

- Pareto optimality (PO)
  - Allocation A is Pareto optimal if there is no allocation B such that $v_i(B_i) \geq v_i(A_i)$ for all agents $i \in N$, and at least one inequality is strict.
  - "Can't make someone happier without making someone else less happy"

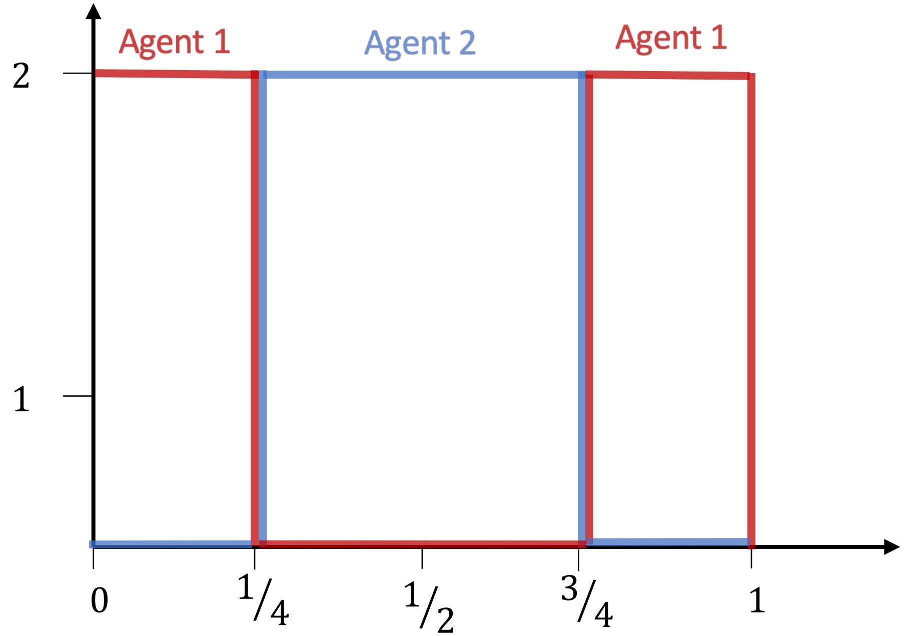
Pareto Optimality (PO)

- Definition
  - We say that an allocation $A = (A_1, \ldots, A_n)$ is PO if there is no alternative allocation $B = (B_1, \ldots, B_n)$ 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: $V_i(B_i) > V_i(A_i), \exists i \in N$

- Q: Is it PO to give the entire cake to agent 1?
  - A: Not necessarily. But yes, if agent 1 values every part of the cake positively.
  - But a "sequential dictatorship" is always Pareto optimal
    - Let agent 1 take whatever she values positively
    - From the rest, let agent 2 take whatever she values positively
    - And so on…

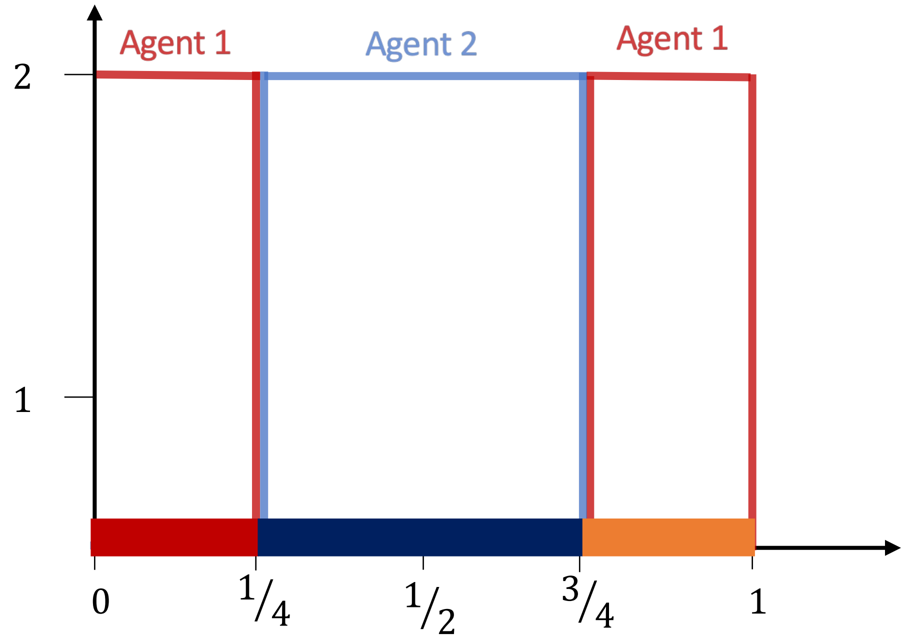
PO+EF+EQ

- Theorem [Barbanel and Brams, 2011]
  With two agents, there is no a simple allocation that is envy-free, equitable, and Pareto optimal.



PO+EF+EQ

- Theorem [Barbanel and Brams, 2011]
  With two agents, there is no an allocation that is envy-free, equitable, and Pareto optimal.
- Reorder each bit g of the cake by $v_1(g)/v_2(g)$
- Choose x such that $v_1([0, x]) = v_2([x, 0])$



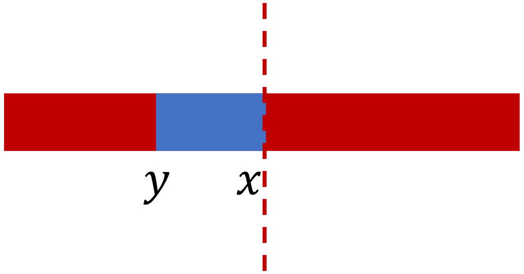
PO+EF+EQ

- Theorem [Barbanel and Brams, 2011]
  With two agents, there is no an allocation that is envy-free, equitable, and Pareto optimal.
- Reorder each bit g of the cake by $v_1(g)/v_2(g)$
- Choose x such that $v_1([0,x]) = v_2([x,0])$

- Question:
  - Why is the algorithm efficient and EF?

PO+EF+EQ

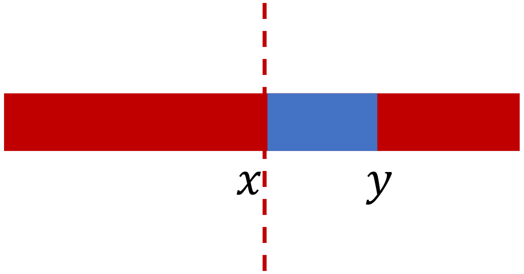
- Efficiency:



  - Case I:
    - Suppose for contradiction that $v_1([y,x]) = 0$ and $v_2([y,x]) > 0$
    - Then, $v_1([y,1]) = 0 \Longrightarrow v_1([0,x]) = v_2([x,y]) = 1$

PO+EF+EQ

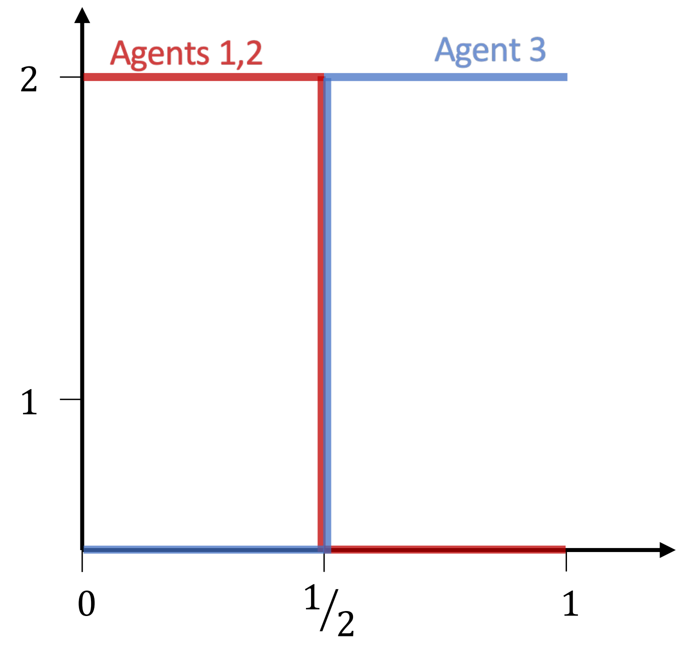
- Efficiency:



  - Case I:
    - Suppose for contradiction that $v_1([y,x]) = 0$ and $v_2([y,x]) > 0$
    - Then, $v_1([y,1]) = 0 \Longrightarrow v_1([0,x]) = v_2([x,y]) = 1$
  - Case II:
    - Suppose for contradiction that $v_2([x,y]) = 0$ and $v_1([x,y]) > 0$
    - Then, $v_2([0,1]) = 0 \Longrightarrow v_1([0,x]) = v_2([x,y]) = 1$
- EF:
  - Suppose that $v_1([0,x]) = v_2([x,0]) < 0.5$
  - But then efficiency is violated!

PO+EF+EQ: (Non-)Existence

- With $n \geq 3$ agents, PO+EQ is impossible



PO + EF

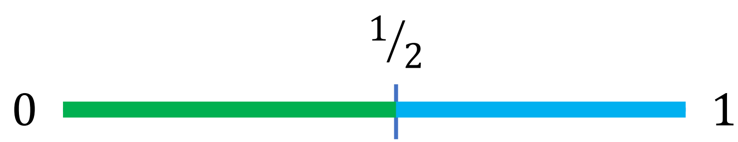
- Theorem [Weller '85]:
  - There always exists an allocation of the cake that is both envy-free and Pareto optimal.

- 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 somewhat non-trivial.
  - Named after John Nash
    - Nash social welfare = product of utilities
    - Different from utilitarian social welfare = sum of utilities

Nash-Optimal Allocation

- Example:
  - Green agent has value 1 distributed over $[0, {}^2/_3]$
  - Blue agent has value 1 distributed over $[0,1]$
  - Without loss of generality (why?) suppose:
    - Green agent gets x fraction of $[0, {}^2/_3]$
    - Blue agent 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 $\frac{3}{4}$

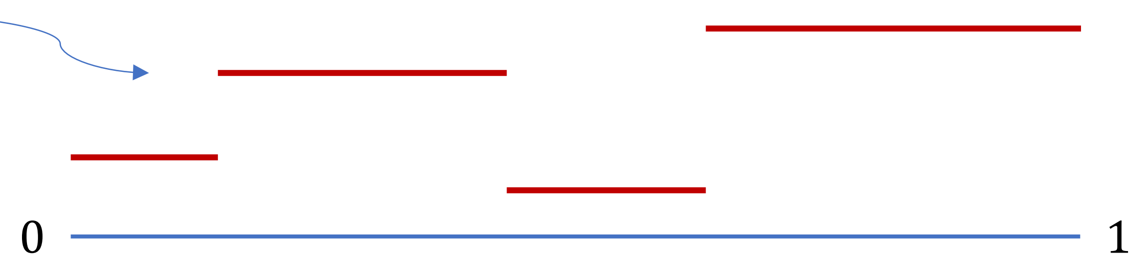
Blue has utility $\frac{1}{2}$

Problem with Nash Solution

- Computing any Pareto optimal allocation already requires an unbounded number of queries

- 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



Strategyproofness

Strategyproofness (SP)

- Direct-revelation mechanisms
  - A direct-revelation mechanism h takes as input all the valuation functions $v_1, \dots, v_n$, and returns an allocation A
  - Notation: $h(v_1, \dots, v_n) = A$, $h_i(v_1, \dots, v_n) = A_i$

- Strategyproofness (deterministic mechanisms)
  - A direct-revelation mechanism h is called strategyproof if

    $\forall v_1, \dots, v_n, \forall i, \forall v_i' : v_i\big(h_i(v_1, \dots, v_n)\big) \geq v_i(h_i(v_1, \dots, v_i', \dots, v_n))$
  - That is, no agent i can achieve a higher value by misreporting her valuation, regardless of what the other agents report

Strategyproofness (SP)

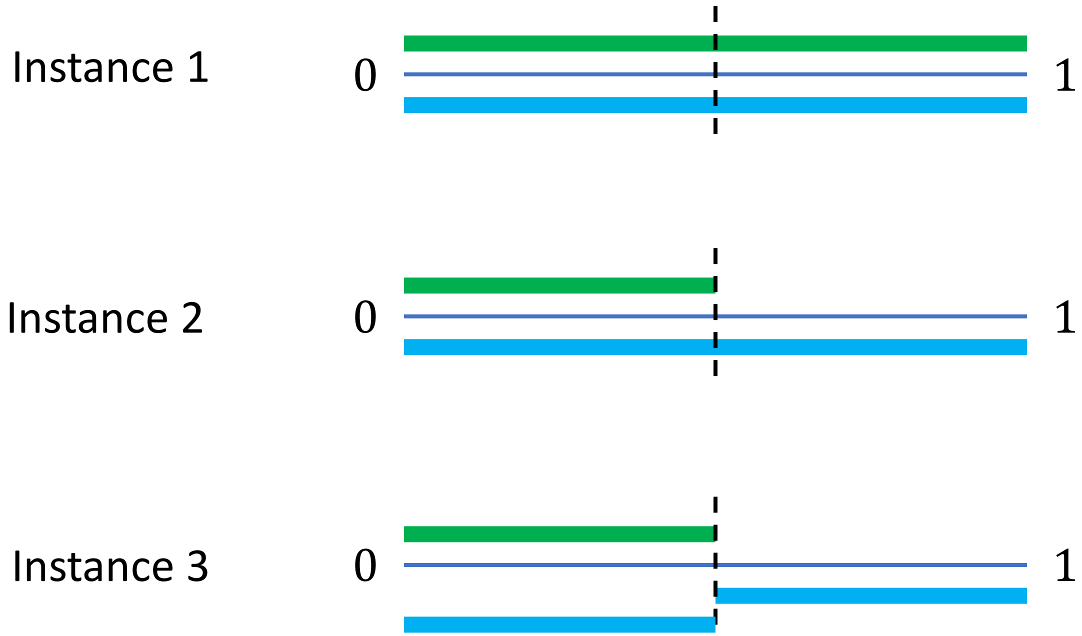
- Strategyproofness (randomized mechanisms)
  - Technically, referred to as "truthfulness-in-expectation"
    - When referring to SP for randomized mechanisms, we will refer to this concept
  - A randomized direct-revelation mechanism h is called strategyproof if

$$\forall v_1, \dots, v_n, \forall i, \forall v_i' : E\left[v_i\left(h_i(v_1, \dots, v_n)\right)\right] \geq E\left[v_i\left(h_i(v_1, \dots, v_i', \dots, v_n)\right)\right]$$

  - That is, no agent i can achieve a higher *expected* value by misreporting her valuation, regardless of what the other agents report
    - Expectation is over the randomness of the mechanism

Deterministic SP Mechanisms

- Theorem [Menon and Larson ’17, Bei et al. ‘17]
  No non-wasteful deterministic SP mechanism is (even approximately) proportional.
- Proof



Deterministic SP Mechanisms

- Theorem [Menon and Larson '17, Bei et al. '17]
  No non-wasteful deterministic SP mechanism is (even approximately) proportional.
  - Since EF is at least as strict as Prop, SP+EF is also impossible subject to non-wastefulness.
  - Non-wastefulness can be replaced by a requirement of "connected pieces", and the impossibility result still holds.

Open Problem

Does the SP+Prop impossibility hold
even without the non-wastefulness assumption?

Deterministic SP Mechanisms

- SP+PO is easy to achieve
  - E.g. serial dictatorship

- SP+PO+EQ is impossible
  - We saw that even EQ+PO allocations may not exist

Open Problem

Does there exist a direct revelation, deterministic SP+EQ mechanism?

Randomized SP Mechanisms

- We want the mechanism *always* return an allocation satisfying a subset of {EQ,EF,PO}, and be SP in expected utilities

- Recall: PO+EQ allocations may not exist
  - Hence, we can only hope for SP+PO+EF or SP+EF+EQ
  - The first is an open problem, but the second combination is achievable!

Open Problem

Does there exist a randomized SP mechanism which always returns a PO+EF allocation?

Randomized SP Mechanisms

- Theorem [Mossel and Tamuz, 2010; Chen et al. 2013]
  There is a randomized SP mechanism that *always* returns an EF+EQ allocation.
  - Recall: In a perfect partition B, $v_i(B_k) = {}^1/_n$ for all $i, k \in N$
  - Algorithm: Compute a perfect partition and return allocation A which randomly assigns the n pieces to the n agents
  - SP: Regardless of what the agents report, agent i receives each piece of the cake with probability $1/n$, and thus has expected value exactly $1/n$
  - EF: Assuming agents report truthfully (due to SP), agent i always receives a cake she values at $1/n$, and according to her, so do others.

Existential Summary

✗ = Impossibility
✓ = Possibility

SP+PO+EF+EQ
- ✗ Rand

SP+PO+EF
- ✗ Det
- ? Rand

SP+PO+EQ
- ✗ Rand

SP+EF+EQ
- ✗ Det
- ✓ Rand

PO+EF+EQ
- ✗ Rand

SP+PO
- ✓ Det

SP+EF
- ✗ Det
- ✓ Rand

SP+EQ
- ? Det
- ✓ Rand

PO+EF
- ✓ Det

PO+EQ
- ✗ Rand

EF+EQ
- ✓ Det

Perfect Partition

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

- 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

- Unfortunately, computing a perfect partition needs an unbounded number of RW queries

Perfect Partition

- If you're given an algorithm for finding a perfect partition…
  - Can you use it to design a randomized protocol that *always* returns an EF allocation and is SP-in-expectation?

  - Yes! Compute a perfect partition and assign the n bundles to the n agents uniformly at random

  - Why is this *always* 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

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

Homogeneous Divisible Goods

- Nash-optimal solution:

  Maximize $\sum_i \log U_i$

  $U_i = \Sigma_g \, x_{i,g} * v_{i,g} \qquad \forall i$

  $\Sigma_i \, x_{i,g} = 1 \qquad \forall g$

  $x_{i,g} \in [0,1] \qquad \forall i, g$

- This is known as the Gale-Eisenberg convex program
  - Can be solved *exactly* in strongly polynomial time

Price of Fairness

Price of Fairness

- Measures the worst-case loss in social welfare due to requirement of a fairness property X
- Social welfare of allocation A is the sum of values of the agents
  - Denoted $sw(A) = \sum_{i \in N} v_i(A_i)$
- Let $\mathcal{F}$ denote the set of feasible allocations and $\mathcal{F}_X$ denote the set of feasible allocations satisfying property X

$$PoF_X = \sup_{v_1,\ldots,v_n} \frac{\max_{A \in \mathcal{F}} sw(A)}{\max_{A \in \mathcal{F}_X} sw(A)}$$

Price of Fairness

- Theorem [Caragiannis et al., 2009]
  For cake-cutting, the price of proportionality is $\Theta(\sqrt{n})$, and the price of equitability is $\Theta(n)$.

- Theorem [Bertsimas et al., 2011]
  For cake-cutting, the price of envy-freeness is also $\Theta(\sqrt{n})$. This is achieved by an allocation maximizing the Nash welfare $\Pi_i \, v_i(A_i)$.