CSC2556

Lecture 7

Fair Division 2: Indivisible Goods Leximin Allocation

Cake-Cutting (contd)
Indivisible Goods

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: $V_i(B_i) > V_i(A_i), \exists i \in N$

  - 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

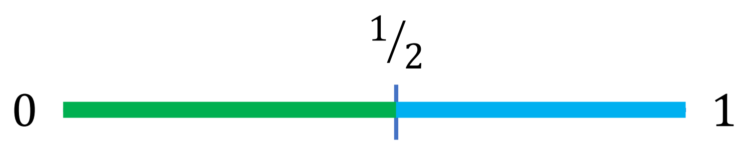
- Theorem [Weller '85]:
  - There always exists an allocation of the cake that is both envy-free and Pareto optimal.

- 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 $\frac{3}{4}$

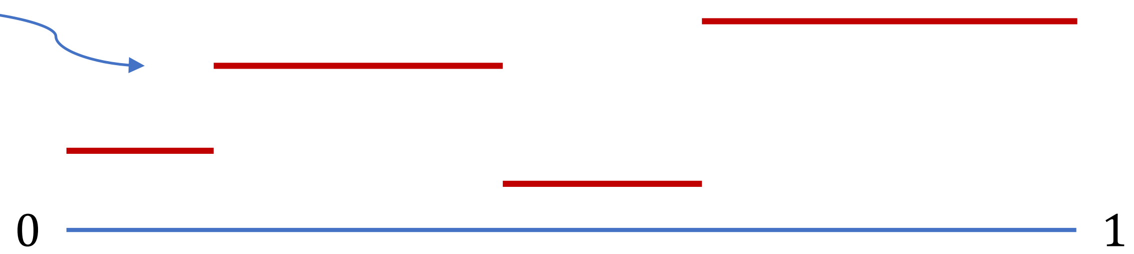
Blue has utility $\frac{1}{2}$

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:

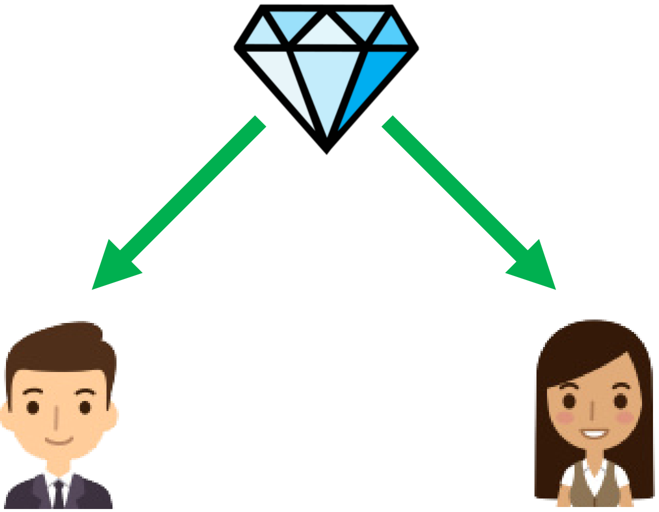
  Maximize $\sum_i \log U_i$

  $$U_i = \Sigma_g \, x_{i,g} * v_{i,g} \qquad \forall i$$
  $$\Sigma_i \, x_{i,g} = 1 \qquad \forall g$$
  $$x_{i,g} \in [0,1] \qquad \forall i, g$$

- 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!



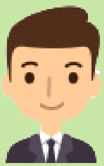
Indivisible Goods: Setting

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

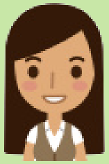
Given such a matrix of numbers, assign each good to a player.

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

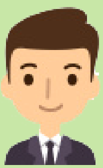
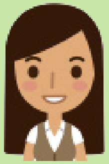
Indivisible Goods

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

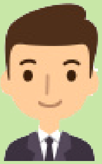
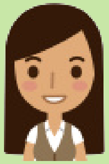
Indivisible Goods

				
	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?

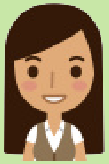
EF1

- Yes! We can use Round Robin.
  - Agents take turns in cyclic order: $1,2, \ldots, n, 1,2, \ldots, n, \ldots$
  - 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.

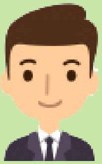
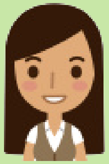
EF1+PO?

- 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 $\text{argmax}_A \prod_{i \in S} V_i(A_i)$

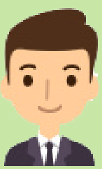
Integral Nash Allocation

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

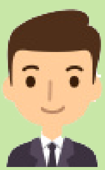
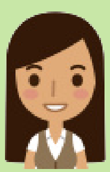
$20 * 8 * (9+10) = 3040$

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

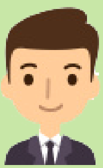
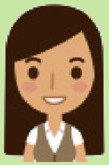
$$(8+7) * 8 * 18 = 2160$$

				
	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.

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

  - 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: $\forall i, j \in N, \exists g \in A_j : \; V_i(A_i) \geq V_i\left(A_j \backslash \{g\}\right)$
  - Intuitively, i doesn't envy j if she gets to remove her most valued item from j's bundle.
- EFx: $\forall i, j \in N, \forall g \in A_j : \; V_i(A_i) \geq V_i\left(A_j \backslash \{g\}\right)$
  - 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, exponentially many CE.

Allocation of Bads

- Indivisible bads
  - EF1: $\forall i, j \;\; \exists b \in A_i \;\; c_i(A_i \backslash \{b\}) \le c_i(A_j)$
  - EFx: $\forall i, j \;\; \forall b \in A_i \; c_i(A_i \backslash \{b\}) \le c_i(A_j)$
    - 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

Leximin (DRF)

Computational Resources

- Resources: Homogeneous divisible resources like CPU, RAM, or network bandwidth

- Valuations: Each player wants the resources in a fixed proportion (Leontief preferences)

- Example:
  - Player 1 requires (2 CPU, 1 RAM) for each copy of task
  - Indifferent between (4,2) and (5,2), but prefers (5,2.5)
  - "fractional" copies are allowed

Model

- Set of players $N = \{1, \dots, n\}$
- Set of resources R, $|R| = m$

- Demand of player i is $d_i = (d_{i1}, \dots, d_{im})$
  - $0 < d_{ir} \le 1$ for every r, $d_{ir} = 1$ for some r
    - "For every 1% of the total available CPU you give me, I need 0.5% of the total available RAM"

- Allocation: $A_i = (A_{i1}, \dots, A_{im})$ where A_{ir} is the fraction of available resource r allocated to i
  - Utility to player i : $u_i(A_i) = \min_{r \in R} A_{ir}/d_{ir}$.
  - We'll assume a non-wasteful allocation
    - Allocates resources proportionally to the demand.

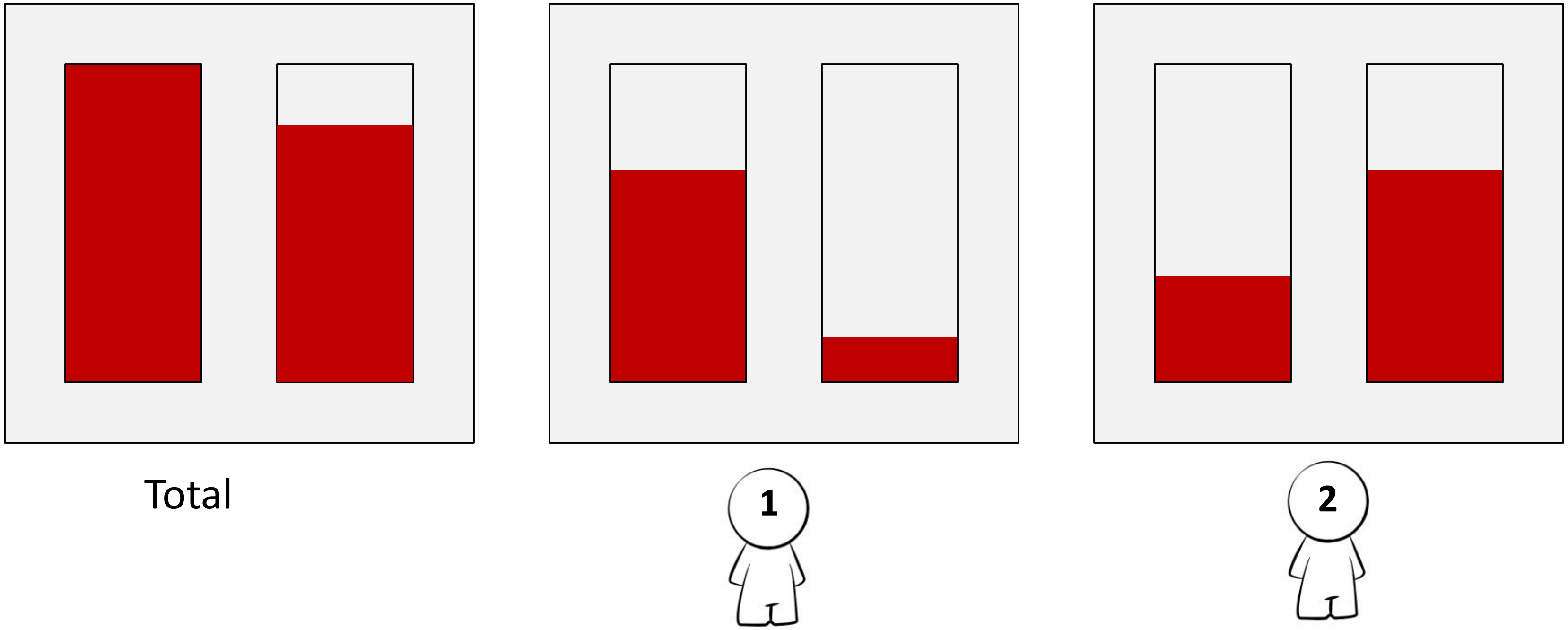
Dominant Resource Fairness

- Dominant resource of i is r such that $d_{ir} = 1$

- Dominant share of i is A_{ir}, where r = dominant resource of i

- Dominant Resource Fairness (DRF) Mechanism
  - Allocate maximal resources while maintaining equal dominant shares.

DRF animated



Properties of DRF

- Envy-free: $u_i(A_i) \geq u_i(A_j), \forall i, j$
  - Why? [Note: EF no longer implies proportionality.]
- Proportionality: $u_i(A_i) \geq 1/n, \forall i$
  - Why?
- Pareto optimality (Why?)
- Group strategyproofness:
  - If a group of players manipulate, it can't be that none of them lose, and at least one of them gains.
  - We'll skip this proof.

The Leximin Mechanism

- Generalizes the DRF Mechanism

- Mechanism:
  - Choose an allocation A that
    - Maximizes $\min_i u_i(A_i)$
    - Among all minimizers, breaks ties in favor of higher second minimum utility.
    - Among all minimizers, breaks ties in favor of higher third minimum utility.
    - And so on…

- Maximizes the egalitarian welfare

The Leximin Mechanism

- DRF is the leximin mechanism
  - In the previous illustration, we didn't need tie-breaking because we assumed $d_{ir} > 0$ for every $i \in N, r \in R$.
  - In practice, not all the players need all the resources.
  - When $d_{ir} = 0$ is allowed, we need to continue allocating even after some agents are saturated.
    - Not all agents have equal dominant shares in the end.
- Theorem [Parkes, Procaccia, S '12]:
  - When $d_{ir} = 0$ is allowed, the leximin mechanism still retains all four properties (proportionality, envy-freeness, Pareto optimality, group strategyproofness).

A Note on Dynamic Settings

- We assumed that all agents are present from the start, and we want a one-shot allocation.
- Real-life environments are dynamic. Agents arrive and depart, and their demands change over time.

- Theorem [Kash, Procaccia, S '14]:
  - A dynamic version of the leximin mechanism satisfies proportionality, Pareto optimality, and strategyproofness along with a relaxed version of envy-freeness when agents arrive one-by-one.

A Note on Dynamic Settings

- Dynamic mechanism design
  - Designing fair, efficient, and game-theoretic mechanisms in dynamic environments is a relatively new research area, and we do not know much.
  - E.g., what if agents can depart, demands can change over time, or agents can submit and withdraw multiple jobs over time?
  - Lots of open questions!

Leximin (Dichotomous Matching)

Matching + Dichotomous Prefs

- Recall the stable matching setting of matching n men to n women.
  - We assumed ranked preferences, and showed that the Gale-Shapley algorithm produces a stable matching.
  - What if agent preferences weren't ranked?
- Suppose the men and women have dichotomous preferences over each other.
  - Each man finds a subset of women "acceptable" (utility 1), and the rest "unacceptable" (utility 0).
  - Same for women's preferences over men.

Matching + Dichotomous Prefs

- Dichotomous preferences induce a bipartite graph betwee men and women.
  - If a perfect matching exists, it’s awesome.
  - What if there is no perfect matching?
    - Any deterministic matching unfairly gives 0 utility to some agents.
    - Solution: randomize!

- Under a random matching, utility to an agent = probability of being matched to an acceptable partner.

Matching + Dichotomous Prefs

- (Integral) Matching:
  - “Select” or “not select” each edge such that the number of selected edges incident on each vertex is at most 1.
- Fractional Matchings:
  - “Put a weight” on each edge such that the total weight of edges incident on each vertex is at most 1.
- Birkoff von-Neumann Theorem:
  - Every fractional matching can be “implemented” as a probability distribution over integral matchings.

Matching + Dichotomous Prefs

- Randomized leximin mechanism:
  - Compute the leximin fractional matching, and implement it as a distribution over integral matchings.
  - Both steps are doable in polynomial time!
- Theorem [Bogomolnaia, Moulin '04]:
  - The randomized leximin mechanism satisfies proportionality, envy-freeness, Pareto optimality, and group-strategyproofness (for both sides).
- In contrast: For ranked preferences, no algorithm can be strategyproof for both sides.

Matching with Capacities

- Proposition 39 in California
  - “Unused resources in public schools should be *fairly* allocated to local charter schools that desire them.”
- Each charter school (agent) i wants d_i unused classrooms at one of the acceptable public schools (facilities) F_i.
  - If the demand is met, the charter school can relocate to the public school facility.
- Each facility j has c_j unused classrooms.
  - We assume facilities don’t have preferences over agents.

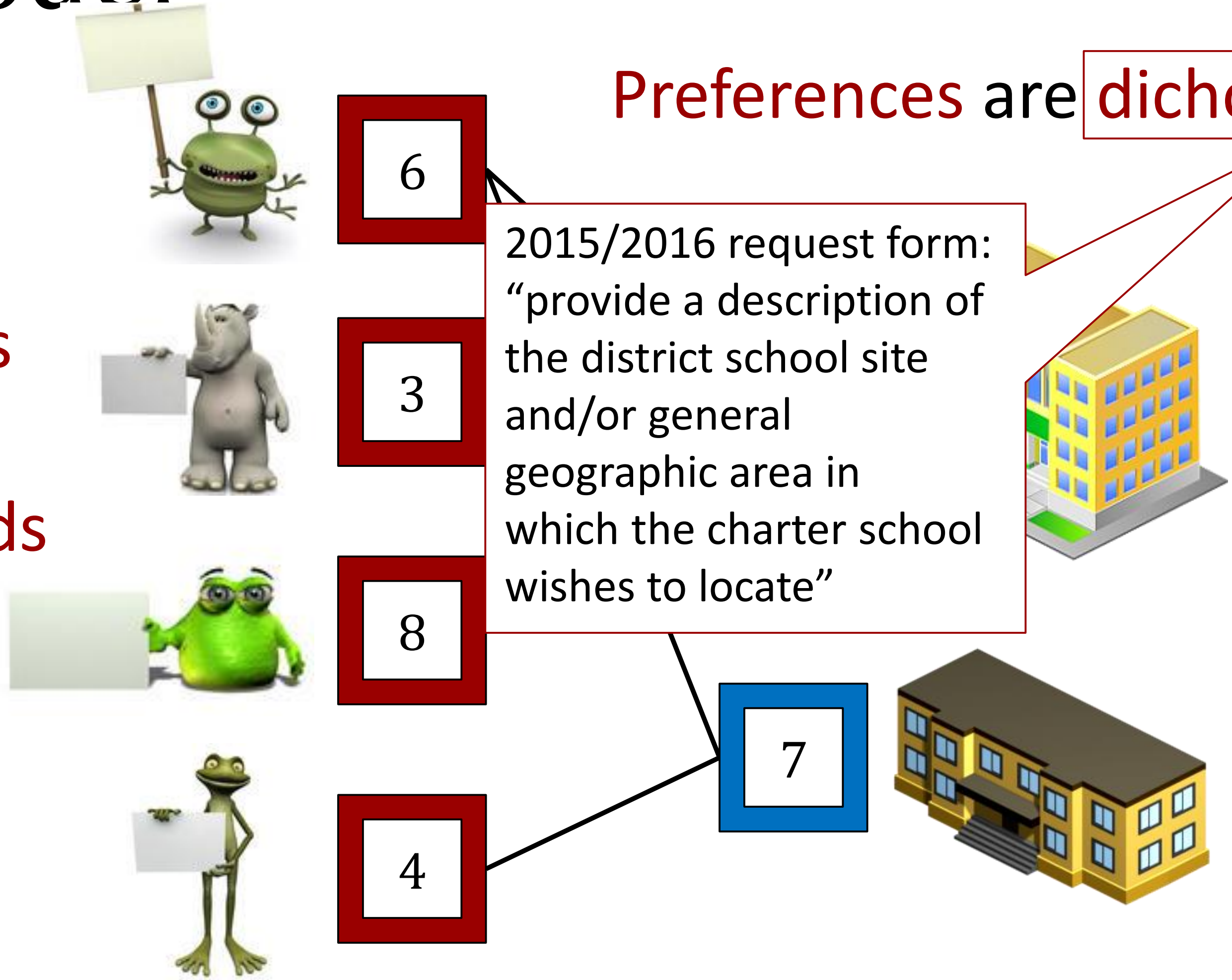
Leximin (Classroom Allocation)

Model

Preferences are dichotomous

2015/2016 request form: “provide a description of the district school site and/or general geographic area in which the charter school wishes to locate”

Agents have demands



Facilities have capacities

Number of unused classrooms

Leximin Strikes Again

- Utility of agent i under a randomized allocation = probability of being allocated d_i classrooms at one of the facilities in F_i.

- Theorem [Kurokawa, Procaccia, S '15]:
  - The randomized leximin mechanism satisfies proportionality, envy-freeness, Pareto optimality, and group strategyproofness.

- Computing this allocation is NP-hard.
  - Unlike DRF and matching under dichotomous preferences.

Leximin Strikes Again

- The result holds in a generic domain which satisfies:
  - Convexity: If two utility vectors are feasible, then so should be their convex combinations.
    - Holds if fractional or randomized allocations are allowed.
  - Equality: The maximum utility of each agent should be the same.
    - Normalize utilities.
  - Shifting Allocations: Swapping allocations of two agents should be allowed.
  - Maximal Utilization: No agent should have a higher utility for agent i's allocation than agent i has.
    - This should hold after the normalization. This is the most restrictive assumption.
- Captures DRF, matching with dichotomous preferences, classroom allocation, and many other settings from the literature.