CS 405/6001: Game Theory and Algorithmic Mechanism Design **Jul–Nov 2026**

Lecture 18: The Uniform Rule and Mechanisms with Money

Lecturer: Swaprava Nath *Scribe(s): Claude Scribe*

Disclaimer: *These notes aggregate content from several texts and have not been subjected to the usual scrutiny deserved by formal publications. If you find errors, please bring to the notice of the Instructor.*

18.1 Recap

A gig platform splits one unit of delivery load among n riders. Rider i gets $s_i \in [0,1]$, the shares sum to 1, and each rider's preference is single-peaked in her own share with peak p_i. Last lecture we showed that an allocation is Pareto efficient exactly when every rider stays on the same side of her peak (Theorem 17.6). Today we look for a rule that is also anonymous and strategyproof. Then we bring money into the picture.

18.2 Anonymity and Strategyproofness for Shares

In this domain, renaming the riders should simply rename their shares.

Definition 18.1 (Anonymity) *An SCF f is **anonymous** (ANON) if for every permutation $\sigma : N \to N$ of the agents,*

$$f_{\sigma(j)}(P^\sigma) = f_j(P), \qquad \forall j \in N.$$

Example 18.2 *Let $N = \{1,2,3\}$ and $\sigma(1) = 2$, $\sigma(2) = 3$, $\sigma(3) = 1$. If the peaks are $P = (0.7, 0.4, 0.3)$, then $P^\sigma = (0.3, 0.7, 0.4)$. Anonymity requires*

$$f_1(0.7,0.4,0.3) = f_2(0.3,0.7,0.4), \quad f_2(0.7,0.4,0.3) = f_3(0.3,0.7,0.4), \quad f_3(0.7,0.4,0.3) = f_1(0.3,0.7,0.4).$$

The rider who asked for 0.7 *gets the same load, whatever her name.*

An SCF f is **manipulable** if there are an agent i, a profile P, and a report P_i' with $f(P_i', P_{-i})\ P_i\ f(P_i, P_{-i})$.

Definition 18.3 (Strategyproofness) *An SCF is **strategyproof** (SP) if it is not manipulable by any agent at any profile.*

Since a rider only cares about her own share, an equivalent statement is: for all $i \in N$, all $P_i, P_i' \in T$, and all $P_{-i} \in T^{n-1}$,

$$f(P_i, P_{-i})\ P_i\ f(P_i', P_{-i}) \quad \text{or} \quad f_i(P_i, P_{-i}) = f_i(P_i', P_{-i}).$$

18.3 Two Candidate Rules

Definition 18.4 (Serial dictatorship) *Fix an order of the agents in advance. Each agent in turn gets either her peak share or whatever is left. If $\sum_{i \in N} p_i < 1$, the last agent is given the leftover share.*

Example 18.5 *Peaks $p_1 = 0.4$, $p_2 = 0.3$, $p_3 = 0.2$, order $1, 2, 3$. Rider 1 gets 0.4, rider 2 gets 0.3, and rider 3 gets the remaining 0.3, even though her peak is 0.2.*

Serial dictatorship is **PE**: the early riders get their peaks, and no reallocation can help the late riders without hurting an early one. It is **SP**: a rider who gets her peak cannot do better, and a rider who does not can only get what is left, which her report does not change. It is **not ANON**: permuting the agents changes who is stuck with the leftover. It is also quite unfair to the last rider.

Definition 18.6 (Proportional rule) *Every agent gets c times her peak, where $c \sum_{i \in N} p_i = 1$.*

The proportional rule is PE and ANON: it depends only on the reported peaks, and every rider is scaled in the same direction. It is **not SP**. Let the peaks be $0.2, 0.3, 0.1$. Then $c = 1/0.6$, and rider 1 gets $1/3$, more than her peak 0.2. If rider 1 reports 0.1 instead, then $c = 1/0.5$ and she gets exactly 0.2. In general, if $\sum_j p_j < 1$ a rider gains by under-reporting her peak, and if $\sum_j p_j > 1$ she gains by over-reporting it.

18.4 The Uniform Rule

How can we get PE, ANON, and SP together? Think of a common water level.

Suppose $\sum_i p_i > 1$. Start every share at 0 and raise a common level λ. When a rider's share reaches her peak, freeze it. Stop when the shares sum to 1. For the peaks $(0.7, 0.4, 0.3)$, the level rises to 0.3, where rider 3 freezes with total 0.9. It then rises to 0.35, where the total is $0.35 + 0.35 + 0.3 = 1$. The allocation is $(0.35, 0.35, 0.3)$.

Suppose $\sum_i p_i < 1$. Start every share at 1 (infeasible) and lower a common water level μ. Freeze a rider once the level hits her peak. Stop when the shares sum to 1. Take the peaks $(0.5, 0.2, 0.1)$, whose sum is 0.8. At $\mu = 1$ the total is 3, and at $\mu = 0.75$ it is 2.25. At $\mu = 0.5$ the level reaches rider 1's peak, so she freezes at 0.5, and the total is 1.5. The level keeps falling for riders 2 and 3: at $\mu = 0.35$ the total is 1.2, and at $\mu = 0.25$ it is $0.5 + 0.25 + 0.25 = 1$. The allocation is $(0.5, 0.25, 0.25)$. The proportional rule would give $(0.625, 0.25, 0.125)$ and push rider 1 above her peak. Here she gets exactly 0.5.

Definition 18.7 (Uniform rule) *The **uniform rule** f^u is*

- $f_i^u(P) = p_i$ *if* $\sum_{i \in N} p_i = 1$*;*
- $f_i^u(P) = \max\{p_i, \mu(P)\}$ *if* $\sum_{i \in N} p_i < 1$*, where* $\mu(P)$ *solves* $\sum_{i \in N} \max\{p_i, \mu\} = 1$*;*
- $f_i^u(P) = \min\{p_i, \lambda(P)\}$ *if* $\sum_{i \in N} p_i > 1$*, where* $\lambda(P)$ *solves* $\sum_{i \in N} \min\{p_i, \lambda\} = 1$*.*

Theorem 18.8 (Sprumont 1991) *The uniform rule is ANON, PE, and SP.*

Proof: ANON. The rule depends only on the peaks, not on who reported them.

PE. If $\sum_i p_i = 1$, $f_i^u(P) = p_i$. If $\sum_i p_i > 1$, $f_i^u(P) = \min\{p_i, \lambda(P)\} \leqslant p_i$ for all i. If $\sum_i p_i < 1$, $f_i^u(P) = \max\{p_i, \mu(P)\} \geqslant p_i$ for all i. Every rider stays on the same side of her peak, so the allocation is PE by Theorem 17.6.

SP. If $\sum_i p_i = 1$, every rider gets her peak and has no reason to lie. Let $\sum_i p_i < 1$, so $f_i^u(P) \geqslant p_i$ for all i. A rider with $f_i^u(P) = p_i$ already has her peak. Consider a rider with $f_i^u(P) > p_i$, so $f_i^u(P) = \mu(P) > p_i$.

Any report $p_i' \leqslant \mu(P)$ leaves the equation for μ unchanged, so her share stays $\mu(P)$. The only way to change the outcome is to report $p_i' > \mu(P) > p_i$. Her share then becomes $p_i' > \mu(P)$, even further from p_i, which is worse. The case $\sum_i p_i > 1$ is symmetric, with λ in place of μ. ∎

The converse also holds [Spr91].

Theorem 18.9 (Sprumont 1991) *A deterministic SCF in the task allocation domain is SP, PE, and ANON if and only if it is the uniform rule.*

18.4.1 Envy-freeness

Definition 18.10 (Envy-freeness) *An allocation is **envy-free** (EF) if no agent prefers another agent's share to her own: $u_i(f_i(P)) \geqslant u_i(f_j(P))$ for all $i \neq j$.*

The uniform rule is envy-free. If $\sum_i p_i = 1$, everyone gets her peak. If $\sum_i p_i < 1$, rider i could envy only if $f_i^u(P) > p_i$, which means $f_i^u(P) = \mu(P)$. Every other rider gets $f_j^u(P) = \max\{\mu(P), p_j\} \geqslant \mu(P)$, which is even further from p_i. So rider i weakly prefers her own share. The case $\sum_i p_i > 1$ is symmetric.

18.4.2 Computation

The uniform rule can be written as a small linear program. First check which case applies, then solve the matching LP.

Case $\sum_i p_i \leqslant 1$:

$$\max_{x,\,\mu} \ \mu \quad \text{s.t.} \quad x_i \geqslant p_i, \quad x_i \geqslant \mu \ \ \forall i \in N, \quad \sum_{i \in N} x_i = 1.$$

The constraints say $x_i \geqslant \max\{p_i, \mu\}$, so a level μ is feasible exactly when $\sum_i \max\{p_i, \mu\} \leqslant 1$. The left side is continuous and nondecreasing in μ, so the largest feasible level is the one where it equals 1, which is $\mu(P)$. At that level the lower bounds already sum to 1, so every bound is tight and $x_i = \max\{p_i, \mu(P)\} = f_i^u(P)$.

Case $\sum_i p_i \geqslant 1$:

$$\min_{x,\,\lambda} \ \lambda \quad \text{s.t.} \quad x_i \leqslant p_i, \quad x_i \leqslant \lambda \ \ \forall i \in N, \quad \sum_{i \in N} x_i = 1.$$

By the mirror argument, the smallest feasible level is $\lambda(P)$, and the optimal x is $x_i = \min\{p_i, \lambda(P)\} = f_i^u(P)$. The constraint $x_i \geqslant 0$ is not needed, since it holds at the optimum.

If $\sum_i p_i = 1$, both programs force $x = p$. Checking the case takes one sum, and linear programs are solvable in polynomial time [Kar84]. So the uniform rule is polynomial-time computable. (One can also sort the peaks and scan for the level directly, in $O(n \log n)$ time.)

So the uniform rule is SP, PE, ANON, EF, and fast. Natural extensions include several resources at once (CPU, RAM, GPU) and non-linear utilities, but we do not pursue them here.

18.5 Mechanism Design with Transfers

We now allow money. A social choice function is $F : \Theta \to X$, where $\Theta = \Theta_1 \times \cdots \times \Theta_n$ is the space of type profiles and X is the space of outcomes. An outcome $x = (a, \pi) \in X$ now has two parts: an **allocation** $a \in A$, and a **payment** vector $\pi = (\pi_1, \ldots, \pi_n)$ with $\pi_i \in \mathbb{R}$.

Examples of allocations:

1. **Public decision.** The campus builds a data center or a park: $a \in \{\text{park}, \text{data center}, \dots\}$.
2. **Divisible good.** Shares of a 5G spectrum band: $a = (a_1, \dots, a_n)$, $a_i \in [0,1]$, $\sum_{i \in N} a_i = 1$.
3. **One indivisible object.** One week on a GPU pod, auctioned: $a_i \in \{0,1\}$, $\sum_{i \in N} a_i \leqslant 1$.
4. **Partitioning objects.** A set S of cloud VMs: $a = (S_1, \dots, S_n)$ with $S_i \subseteq S$ for all $i \in N$ and $S_i \cap S_j = \emptyset$ for all $i \neq j$.

Types and valuations. Agent i's type $\theta_i \in \Theta_i$ is her private information. Her benefit is given by a **valuation function** $v_i : A \times \Theta_i \to \mathbb{R}$, which depends only on her own type (**independent private values**). For the campus decision, an environmentalist type has $v_i(D, \theta_i^{\text{env}}) < v_i(P, \theta_i^{\text{env}})$, while a business-minded type, say a startup founder, has $v_i(D, \theta_i^{\text{bus}}) > v_i(P, \theta_i^{\text{bus}})$.

Payments. If $\pi_i > 0$, agent i pays π_i. If $\pi_i < 0$, she receives $|\pi_i|$. If $\pi_i = 0$, no money moves.

Definition 18.11 (Quasi-linear utility) *When the outcome is $x = (a, \pi)$ and agent i has type θ_i, her utility is*

$$u_i((a, \pi), \theta_i) = v_i(a, \theta_i) - \pi_i.$$

*Types that evaluate outcomes this way form the **quasi-linear domain**.*

Proposition 18.12 *In the quasi-linear domain, if two outcomes have the same allocation and $\pi_i' < \pi_i$, then every type of agent i strictly prefers (a, π') to (a, π).*

Proof: $u_i((a, \pi'), \theta_i) = v_i(a, \theta_i) - \pi_i' > v_i(a, \theta_i) - \pi_i = u_i((a, \pi), \theta_i)$ for every $\theta_i \in \Theta_i$. ■

In the unrestricted domain, both rankings of (a, π) and (a, π') would be allowed. So quasi-linearity is a domain restriction. This restriction opens the door to many non-dictatorial mechanisms.

18.5.1 Allocation and payment rules

In the quasi-linear domain, the SCF splits as $F \equiv (f, (p_1, \dots, p_n)) \equiv (f, p)$, with an **allocation rule** $f : \Theta \to A$ and **payment rules** $p_i : \Theta \to \mathbb{R}$ for $i \in N$. Hence

$$F : \Theta_1 \times \cdots \times \Theta_n \to A \times \mathbb{R}^n.$$

Some allocation rules:

1. **Constant:** $f^c(\theta) = a$ for all $\theta \in \Theta$.
2. **Dictatorial:** $f^D(\theta) \in \arg\max_{a \in A} v_d(a, \theta_d)$ for some fixed $d \in N$.
3. **Allocatively efficient (utilitarian):** $f^{AE}(\theta) \in \arg\max_{a \in A} \sum_{i \in N} v_i(a, \theta_i)$. This differs from Pareto efficiency, which is defined on outcomes and so also considers payments.
4. **Affine maximizer:** $f^{AM}(\theta) \in \arg\max_{a \in A} \left(\sum_{i \in N} \lambda_i v_i(a, \theta_i) + \kappa(a) \right)$, with $\lambda_i \geqslant 0$, not all zero. With $\lambda_i = 1$ for all i and $\kappa \equiv 0$ we get f^{AE}. With $\lambda_d = 1$, $\lambda_j = 0$ for $j \neq d$, and $\kappa \equiv 0$ we get f^D.
5. **Max-min (egalitarian):** $f^{MM}(\theta) \in \arg\max_{a \in A} \min_{i \in N} v_i(a, \theta_i)$.

Some payment properties:

1. **No deficit:** $\sum_{i \in N} p_i(\theta) \geqslant 0$ for all θ. The mechanism never loses money.
2. **No subsidy:** $p_i(\theta) \geqslant 0$ for all θ and all i. Nobody is ever paid.
3. **Budget balanced:** $\sum_{i \in N} p_i(\theta) = 0$ for all θ. Money only moves between agents.

Both no subsidy and budget balance imply no deficit.

18.6 Dominant Strategy Incentive Compatibility

Definition 18.13 (DSIC) *A mechanism (f, p) is **dominant strategy incentive compatible** (DSIC) if for every $i \in N$, all $\theta_i, \theta_i' \in \Theta_i$, and all $\tilde{\theta}_{-i} \in \Theta_{-i}$,*

$$v_i(f(\theta_i, \tilde{\theta}_{-i}), \theta_i) - p_i(\theta_i, \tilde{\theta}_{-i}) \geqslant v_i(f(\theta_i', \tilde{\theta}_{-i}), \theta_i) - p_i(\theta_i', \tilde{\theta}_{-i}).$$

DSIC means that truth-telling is a weakly dominant strategy equilibrium. We say the payment rule p **implements** f in dominant strategies, or that f is implementable. In the quasi-linear domain we are often more interested in the allocation rule, and then look for a payment that implements it.

Example 18.14 *Let $N = \{1, 2\}$ and $\Theta_1 = \Theta_2 = \{\theta^H, \theta^L\}$. For player 1, DSIC requires, for all $\theta_2 \in \Theta_2$,*

$$v_1(f(\theta^H, \theta_2), \theta^H) - p_1(\theta^H, \theta_2) \geqslant v_1(f(\theta^L, \theta_2), \theta^H) - p_1(\theta^L, \theta_2),$$
$$v_1(f(\theta^L, \theta_2), \theta^L) - p_1(\theta^L, \theta_2) \geqslant v_1(f(\theta^H, \theta_2), \theta^L) - p_1(\theta^H, \theta_2),$$

and two symmetric inequalities hold for player 2.

Shifting payments. Suppose (f, p) is DSIC and let

$$q_i(\theta_i, \theta_{-i}) = p_i(\theta_i, \theta_{-i}) + h_i(\theta_{-i})$$

for arbitrary functions $h_i : \Theta_{-i} \to \mathbb{R}$. Then (f, q) is also DSIC: in the DSIC inequality for q, the term $h_i(\tilde{\theta}_{-i})$ appears on both sides and cancels. So if one payment implements f, uncountably many do. The converse question, when *all* implementing payments differ only by such a term, comes later. A related fact (exercise): if $f(\theta) = f(\tilde{\theta}_i, \theta_{-i})$ and p implements f, then $p_i(\theta) = p_i(\tilde{\theta}_i, \theta_{-i})$.

18.7 Pareto Optimality and Allocative Efficiency

Definition 18.15 (Pareto optimality) *A mechanism $(f, (p_1, \ldots, p_n))$ is **Pareto optimal** (PO) if for every $\theta \in \Theta$ there is no allocation $b \neq f(\theta)$ and payments $(\pi_1, \ldots, \pi_n)$ with $\sum_{i \in N} \pi_i \geqslant \sum_{i \in N} p_i(\theta)$ such that*

$$v_i(b, \theta_i) - \pi_i \geqslant v_i(f(\theta), \theta_i) - p_i(\theta) \quad \forall i \in N,$$

with strict inequality for some $i \in N$.

The budget condition matters. Without it, Pareto optimality is meaningless: one could always subsidize every agent heavily and make everyone better off. The condition asks whether we can do better without spending more.

Theorem 18.16 *A mechanism $(f, (p_1, \dots, p_n))$ is Pareto optimal if and only if f is allocatively efficient.*

Proof: ($\Leftarrow$) We show $\neg$PO $\Rightarrow$ $\neg$AE. If the mechanism is not PO, there are θ, b, and π with $\sum_i \pi_i \geqslant \sum_i p_i(\theta)$ and $v_i(b, \theta_i) - \pi_i \geqslant v_i(f(\theta), \theta_i) - p_i(\theta)$ for all i, strict for some j. Summing,

$$\sum_{i \in N} v_i(b, \theta_i) - \sum_{i \in N} v_i(f(\theta), \theta_i) > \sum_{i \in N} \pi_i - \sum_{i \in N} p_i(\theta) \geqslant 0,$$

so f is not AE.

($\Rightarrow$) We show $\neg$AE $\Rightarrow$ $\neg$PO. If f is not AE, there are θ and $b \neq f(\theta)$ with $\delta = \sum_i v_i(b, \theta_i) - \sum_i v_i(f(\theta), \theta_i) > 0$. Set

$$\pi_i = v_i(b, \theta_i) - v_i(f(\theta), \theta_i) + p_i(\theta) - \delta/n, \qquad \forall i \in N.$$

Then $(v_i(b, \theta_i) - \pi_i) - (v_i(f(\theta), \theta_i) - p_i(\theta)) = \delta/n > 0$ for every i, and $\sum_i \pi_i = \sum_i p_i(\theta)$. So the mechanism is not PO. ■

The proof splits the extra welfare δ equally. With money, efficiency becomes a question about the allocation rule alone.

18.8 Groves Payments

Can we implement f^{AE} truthfully? Consider

$$p_i^G(\theta_i, \theta_{-i}) = h_i(\theta_{-i}) - \sum_{j \neq i} v_j\big(f^{AE}(\theta_i, \theta_{-i}), \theta_j\big),$$

where $h_i : \Theta_{-i} \to \mathbb{R}$ is arbitrary. This is the **Groves payment**. Apart from h_i, agent i is paid the welfare of everyone else.

Example 18.17 (Four AI labs, one GPU pod for a week) *Let $N = \{1, 2, 3, 4\}$ and one indivisible object. The values are $\theta_1 = 10$, $\theta_2 = 8$, $\theta_3 = 6$, $\theta_4 = 4$ for getting the pod, and 0 otherwise. Let $h_i(\theta_{-i}) = \min \theta_{-i}$. Under truthful reports, $h_1 = 4$, $h_2 = 4$, $h_3 = 4$, $h_4 = 6$. The efficient allocation gives the pod to lab 1. The payments are*

$$p_1 = 4 - 0 = 4, \quad p_2 = 4 - 10 = -6, \quad p_3 = 4 - 10 = -6, \quad p_4 = 6 - 10 = -4.$$

Only the winner pays. The others get paid. Surprisingly, this mechanism is truthful.

Theorem 18.18 *Groves mechanisms are DSIC.*

Proof: Fix player i, reports $\tilde{\theta}_{-i}$ of the others, a true type θ_i, and a lie θ_i'. Let $f^{AE}(\theta_i, \tilde{\theta}_{-i}) = a$ and $f^{AE}(\theta_i', \tilde{\theta}_{-i}) = b$. By the definition of f^{AE},

$$v_i(a, \theta_i) + \sum_{j \neq i} v_j(a, \tilde{\theta}_j) \geqslant v_i(b, \theta_i) + \sum_{j \neq i} v_j(b, \tilde{\theta}_j).$$

Player i's utility from reporting θ_i is

$$\begin{aligned} v_i(f^{AE}(\theta_i, \tilde{\theta}_{-i}), \theta_i) - p_i(\theta_i, \tilde{\theta}_{-i}) &= v_i(a, \theta_i) - h_i(\tilde{\theta}_{-i}) + \sum_{j \neq i} v_j(a, \tilde{\theta}_j) \\ &\geqslant v_i(b, \theta_i) - h_i(\tilde{\theta}_{-i}) + \sum_{j \neq i} v_j(b, \tilde{\theta}_j) \\ &= v_i(f^{AE}(\theta_i', \tilde{\theta}_{-i}), \theta_i) - p_i(\theta_i', \tilde{\theta}_{-i}). \end{aligned}$$

Since i was arbitrary, this holds for all $i \in N$. ■

Groves makes each agent's utility equal to total welfare minus a term she cannot influence. So her goal lines up with the goal of f^{AE}.

18.9 Summary

Serial dictatorship is PE and SP but not anonymous. The proportional rule is PE and anonymous but not SP. The uniform rule is the unique SP, PE, ANON rule for sharing a divisible task, and it is also envy-free and fast to compute. With money, an outcome is an allocation plus a payment, and utilities are quasi-linear. Adding any $h_i(\theta_{-i})$ to a DSIC payment keeps it DSIC. In the quasi-linear domain, Pareto optimality is equivalent to allocative efficiency, and Groves payments implement the efficient allocation in dominant strategies. Next week: the VCG mechanism, a special Groves mechanism.

References

[Kar84] N. Karmarkar, "A new polynomial-time algorithm for linear programming," *Combinatorica*, 4(4), 1984, pp. 373–395.

[Spr91] Y. Sprumont, "The division problem with single-peaked preferences: a characterization of the uniform allocation rule," *Econometrica*, 59(2), 1991, pp. 509–519.