A characterization of 2-player mechanisms for scheduling

ESA'08

Angelina Vidali

University of Athens
Department of Informatics and Telecommunications
http://users.uoa.gr/-avidali

17, September '08

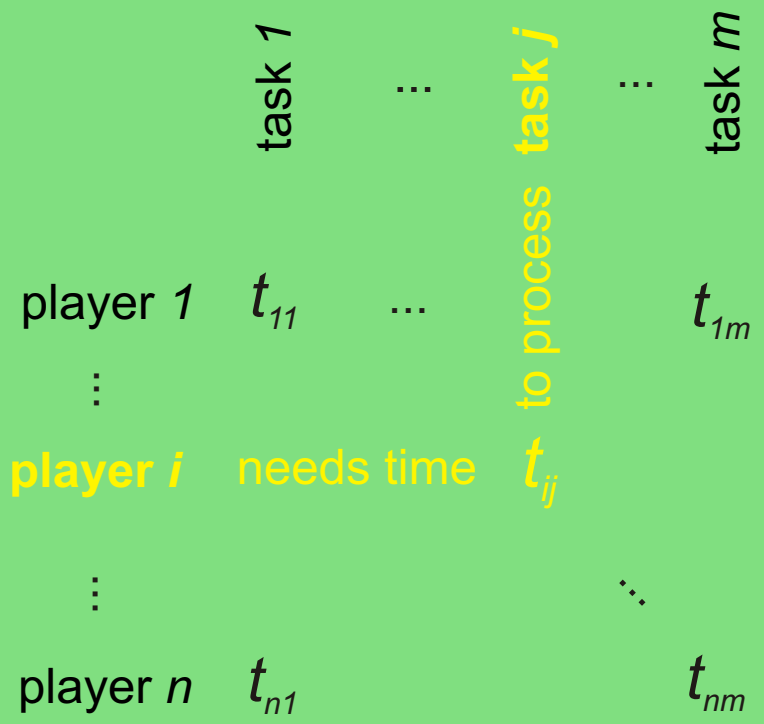
Joint work with:

George Christodoulou (Max Plank Institut für Informatik)
Elias Koutsoupias (University of Athens)

Scheduling unrelated machines

The matrix of processing times

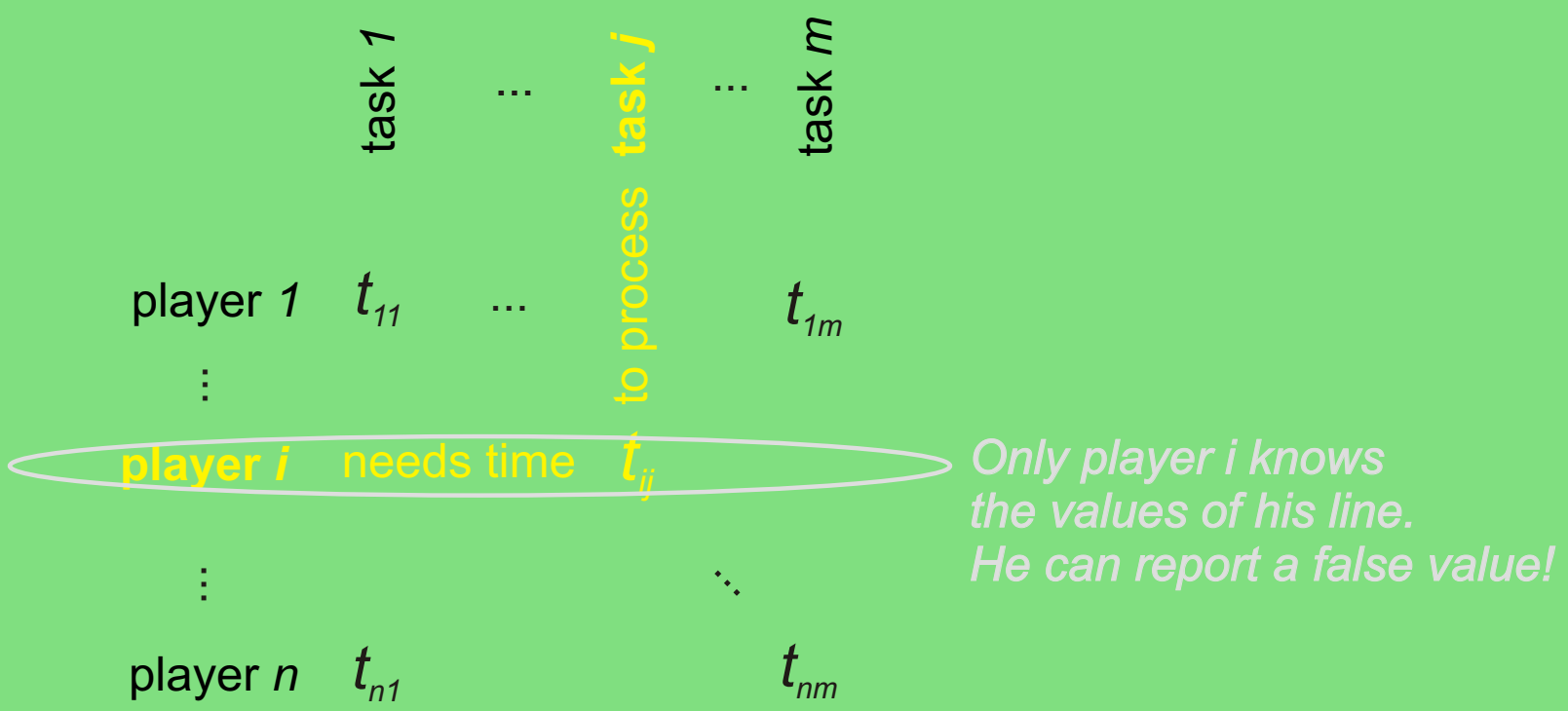
We want to process m tasks using n machines(/selfish players).
We have the following matrix of processing times:



Scheduling unrelated machines

The matrix of processing times

We want to process m tasks using n machines(/selfish players).
We have the following matrix of processing times:



Input and Output

Input	Output
$t = \begin{pmatrix} t_{11} & t_{12} & \cdots & t_{1m} \\ t_{21} & t_{22} & \cdots & t_{2m} \\ \cdots & & & \\ t_{n1} & t_{n2} & \cdots & t_{nm} \end{pmatrix}$	$a = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1m} \\ a_{21} & a_{22} & \cdots & a_{2m} \\ \cdots & & & \\ a_{n1} & a_{n2} & \cdots & a_{nm} \end{pmatrix}$
$t_{ij} \in \mathbb{R}^+$	$a_{ij} \in \{0, 1\}$ $\sum_i a_{ij} = 1$

n machines

m tasks

Scheduling unrelated machines

Protocol

- The players declare their values
- The mechanism allocates all tasks (allocation algorithm)
- The mechanism pays the players based on the declared values and the allocation (payment algorithm)

The objective of each player: utility maximization

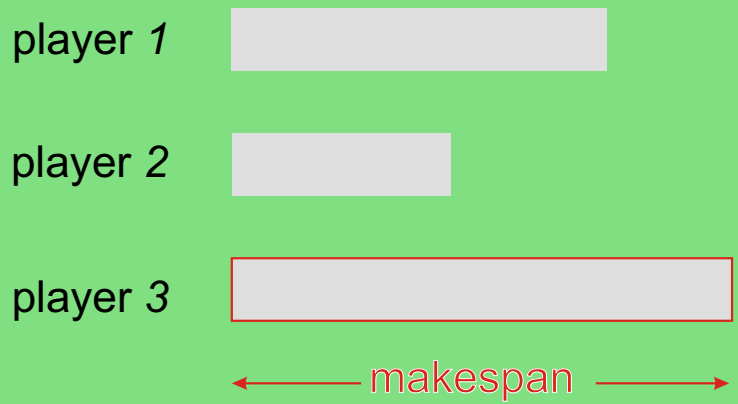
maximize{payment−processing time}

time=money...

MinMax objective of the mechanism designer:

Finish with all tasks as soon as possible! i.e., minimize the makespan

makespan=the time of the player that finishes last



- Tasks allocated to the same player are processed in series.
- Tasks allocated to different players are processed in parallel.

Theorem (Nisan and Ronen 1998)

No mechanism can achieve this objective.

They conjectured the best possible approximation ratio is n. They showed it cannot be better than 2.

Brief history of scheduling unrelated machines

Restricted to the objective of minimizing the makespan

- It is a well-studied NP-hard problem. Lenstra, Shmoys, and Tardos showed that its approximation ratio is between 3/2 and 2.
- Nisan and Ronen in 1998 initiated the study of its mechanism-design version.
  - They gave a mechanism with approximation ratio n.
  - They showed a lower bound of 2.
  - They conjectured that the right answer is n.
  - They also gave a randomized mechanism with approximation ratio 7/4 for 2 players.
- Archer and Tardos considered the related machines problem. In this case, for each machine there is a single value (instead of a vector), its speed.

Recent results

Deterministic

- The lower bound was improved from 2 to 2.41 (Christodoulou - Koutsoupias - Vidali, SODA 2007)
- . . . and then to 2.61 employing $n \longrightarrow \infty$ machines (Koutsoupias - Vidali, MFCS 2007)
- For 2 machines the only truthful mechanisms with bounded approximation ratio are task-independent. (Dobzinski - Sundararajan, EC 2008)

Recent Results [2]

Fractional

- The approximation ratio is between $2 - 1/n$ and $(n+1)/2$ (Christodoulou - Koutsoupias - Kovacs, ICALP 2007)

Randomized

- The approximation ratio is between $2 - 1/n$ and $7/8\,n$ (Mu'alem and Schapira, SODA 2007).
- A 1.67-approximation mechanism for 2 machines [improving on $7/4 = 1.75$ Nisan-Ronen] (Lu - Yu, STACS 2008)

Discrete (only two values: high and low)

Mechanism with approximation ratio 2 (Lavi - Swamy EC 2007).

We can concentrate on truthful mechanisms

Our players are potential liars.

Definition (Truthful mechanisms)

A mechanism is truthful if revealing the true values is dominant strategy of each player.

that is: *"Players have nothing to gain by lying."*

Theorem (The revelation principle)

For every mechanism M there is an equivalent truthful one M'.

that is: "In M' all players achieve the same utility as in M, and without having to lie."

Truthful = Monotone

Definition (Monotonicity Property)

An allocation algorithm is monotone if for every two inputs t and t' which differ only on machine i (i.e., on the i-the row) the associated allocations a and a' satisfy

$$(a_i - a'_i) \cdot (t_i - t'_i) \leq 0,$$

where $\cdot$ denotes the dot product of the vectors.

Theorem (Saks, Lan Yu 2005)

The Monotonicity Property is a necessary and sufficient condition (without any reference to payments!) for truthfulness.

Generalizing this Archer and Kleinberg [EC 2008] determine a necessary and sufficient condition for truthfulness for the case when the outcome set is infinite (like in the fractional case).

The monotonicity property

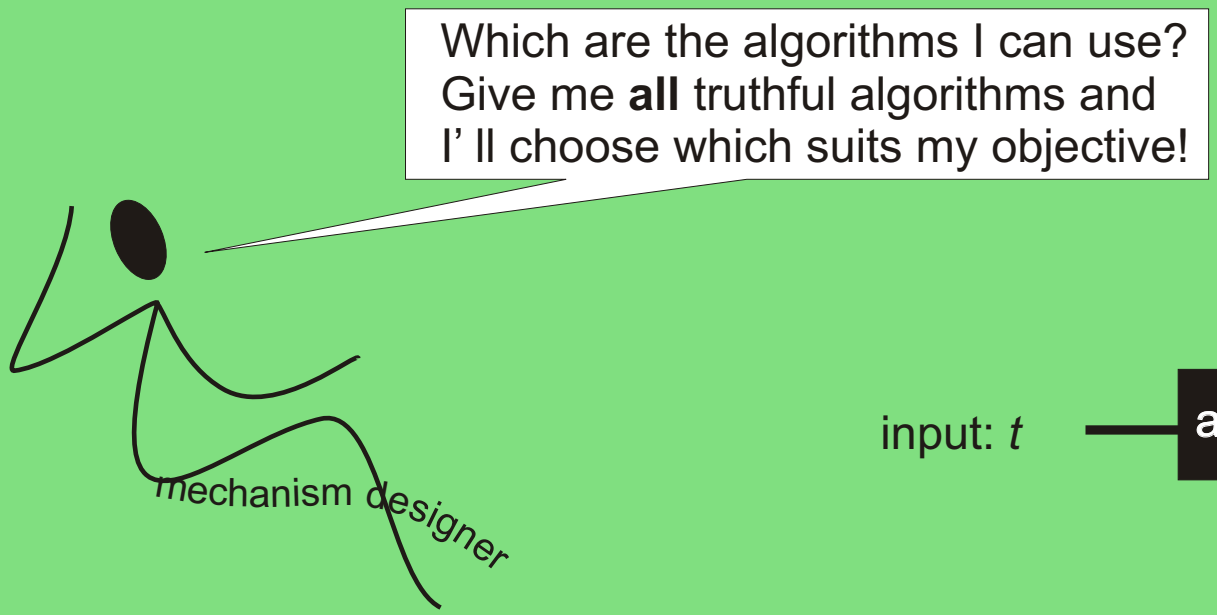
$$(\mathbf{a_i} - \mathbf{a'_i}) \cdot (\mathbf{t_i} - \mathbf{t'_i}) \leq 0,$$

$$\begin{pmatrix} t_{11} & t_{12} & \cdots & t_{1m} \\ \cdots & & & \\ \mathbf{t_{i1}} & \mathbf{t_{i2}} & \cdots & \mathbf{t_{im}} \\ \cdots & & & \\ t_{n1} & t_{n2} & \cdots & t_{nm} \end{pmatrix} \rightarrow \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1m} \\ \cdots & & & \\ \mathbf{a_{i1}} & \mathbf{a_{i2}} & \cdots & \mathbf{a_{im}} \\ \cdots & & & \\ a_{n1} & a_{n2} & \cdots & a_{nm} \end{pmatrix}$$

$$\begin{pmatrix} t_{11} & t_{12} & \cdots & t_{1m} \\ \cdots & & & \\ \mathbf{t'_{i1}} & \mathbf{t'_{i2}} & \cdots & \mathbf{t'_{im}} \\ \cdots & & & \\ t_{n1} & t_{n2} & \cdots & t_{nm} \end{pmatrix} \rightarrow \begin{pmatrix} a'_{11} & a'_{12} & \cdots & a'_{1m} \\ \cdots & & & \\ \mathbf{a'_{i1}} & \mathbf{a'_{i2}} & \cdots & \mathbf{a'_{im}} \\ \cdots & & & \\ a'_{n1} & a'_{n2} & \cdots & a'_{nm} \end{pmatrix}$$

Characterize all truthful algorithms for scheduling!

It is then easy to determine which one has the best approximation ratio...

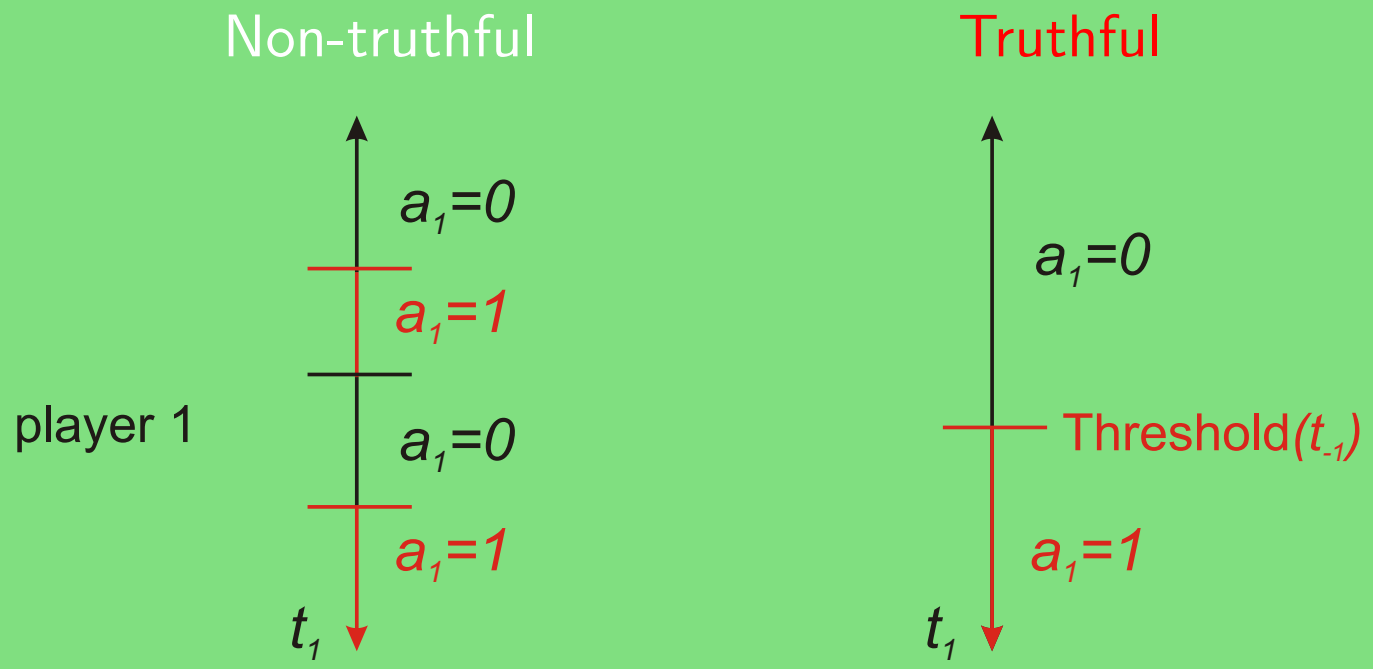


- Monotonicity is a Local Property between each pair of instances (input,output) and (input',output') of the problem.
- We give a Global characterization: Which types of algorithms are allowed for the mechanism designer to use? We only know 2 types of algorithms, can we prove there are no other algorithms?

A Geometrical Interpretation of Monotonicity

Singe-task case

suppose there is only one task. Fix t_{-1} (i.e. the values of all players except for player 1).

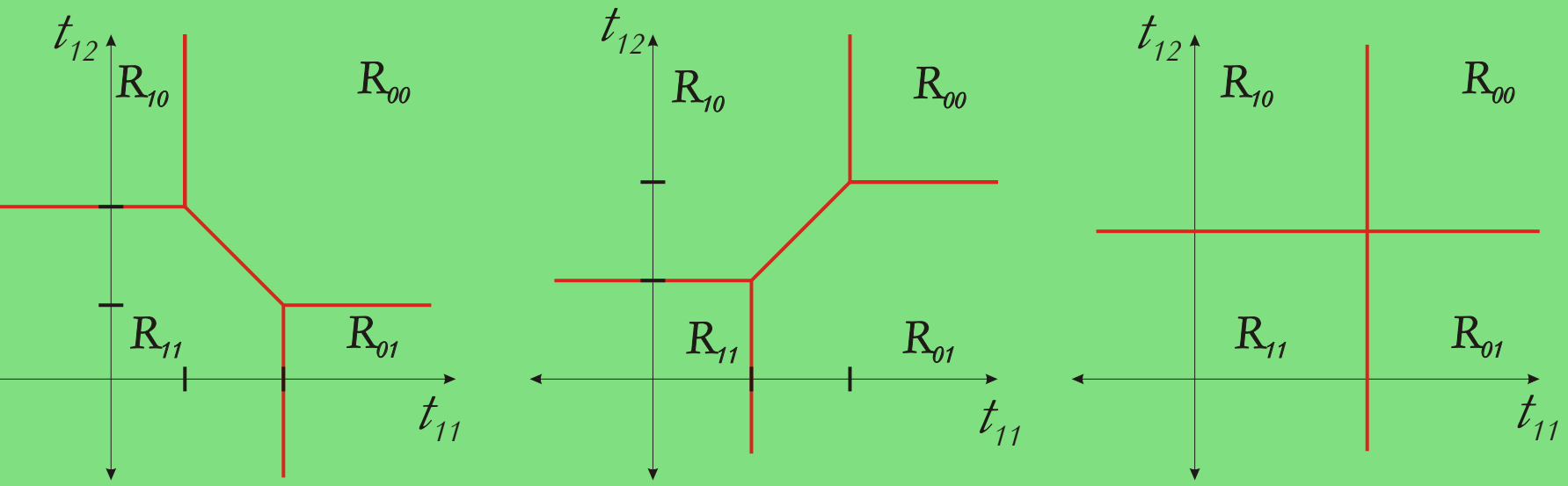


A truthful allocation should be decreasing with respect to the processing time of player 1.

A Geometrical Interpretation of Monotonicity

The case of two tasks

Fix t_{-1} (i.e. the values of all players except for player 1),
R_{10} := the region where player 1 has assignment 10



$(a_1 - a_1')t_{11} + (a_2 - a_2')t_{12} = f_{a:a'}(t_{-1})$ separates $R_a, R_{a'}$

$t_{11} + t_{12} = f_{11:00}(t_{-1})$ separates R_{11} and R_{00}.

Monotonicity determines the slope of the separating hyperplane.

Task-Independent and Threshold Mechanisms

Both have no sloped lines on the projections.

Definition (Task-Independent)

Each task is allocated independently of the rest.

Definition (Threshold Mechanisms)

"Each task is allocated independently of the other values of player i."
There are thresholds $h_{ij}(t_{-i})$ such that the player i gets task j iff the value t_{ij} is less than h_{ij}.

Example

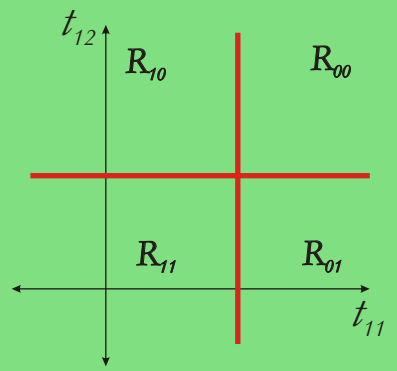
Task-independent:

$$\begin{pmatrix} t_{11} & t_{12} & t_{13} & t_{14} \\ t_{21} & t_{22} & t_{23} & t_{24} \\ t_{31} & t_{32} & t_{33} & t_{34} \end{pmatrix}$$

Threshold:

$$\begin{pmatrix} t_{11} & t_{12} & t_{13} & t_{14} \\ t_{21} & t_{22} & t_{23} & t_{24} \\ t_{31} & t_{32} & t_{33} & t_{34} \end{pmatrix}$$

(The red tasks can affect the allocation a_{12}.)



The VCG (Vickrey-Clark-Grooves) Mechanism

player 1 $\begin{pmatrix}3\\ \mathbf{1}\\ 2\end{pmatrix}$ $\begin{pmatrix}4\\ 8\\ \mathbf{2}\end{pmatrix}$ $\begin{pmatrix}6\\ 3\\ \mathbf{2}\end{pmatrix}$

player 2

player 3

$\begin{pmatrix}3 & 4 & 6\\ \mathbf{1} & 8 & 3\\ 2 & \mathbf{2} & \mathbf{2}\end{pmatrix}$ Minimum sum of processing times $= \mathbf{1} + \mathbf{2} + \mathbf{2}$

- One task: Give the task to the player with the minimum processing time.
- Many tasks: Give each task independently to the player with minimum processing time=Minimize the sum of processing times $\sum_i a_i \cdot t_i$.

VCG and its generalization

Generalizing the VCG to get a sloped line.

Definition (VCG)

The Vickrey-Clarke-Groves (VCG) mechanism selects the outcome which minimizes the sum of the values of the players $\sum_i a_i \cdot t_i$.

Definition (Affine minimizers)

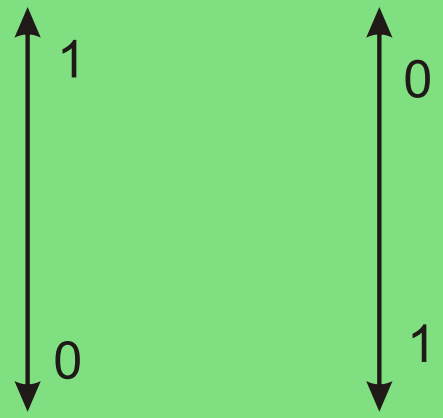
A mechanism is an affine minimizer if there are constants $\lambda_i > 0$ (one for each player i) and γ_a (one for each of the n^m allocations) such that the mechanism selects the allocation a which minimizes $\sum_i \lambda_i a_i \cdot t_i + \gamma_a$.

Example

$$\begin{array}{lllll} \text{player 1 } \lambda_1 a_1 \cdot \rightarrow & (t_{11} & t_{12} & t_{13}) & + \\ \text{player 2 } \lambda_2 a_2 \cdot \rightarrow & (t_{21} & t_{22} & t_{23}) & +\gamma_a \end{array}$$

Auctions or Scheduling?

The world upside down



- Auction: sell the objects to bidder who values them high
- Scheduling: allocate the task to machines with small processing times
  Change max$\longrightarrow$ min

Our objectives are different...
For a characterization it doesn't matter.

Objective of the mechanism designer:

- Auctions: maximize the "total happiness"! The VCG mechanism achieves it, ... but is computationally hard.
- Scheduling: minimize the makespan! The VCG mechanism only gives an n-approximation, ... though computationally easy.

But a characterization gives all thinkable algorithms regardless of objective! For the time being forget the objective.

Having a different domain of valuations matters.

- Urestricted domains: The valuations of player 1 for two different assignments are independent of each other.
- Combinatorial auction domain: a players valuation cannot decrease from taking more items, it depends on what **he** gets, $v_i(\emptyset) = 0$.
- Scheduling domain: The valuations are additive, i.e. If player 3 gets two jobs k, l its valuation(/total processing time) is $t_{3k} + t_{3l}$

The richer the domain, the easier the characterization.

The class of algorithms for scheduling is a superset of the other two classes. In this sense scheduling is more difficult than the other two problems.

Characterizations

Theorem (Roberts, 1979)

For the unrestricted domain with at least 3 outcomes, the only truthful mechanisms are the generalized VCG mechanisms.

Characterizations

Theorem (Roberts, 1979)

For the unrestricted domain with at least 3 outcomes, the only truthful mechanisms are the generalized VCG mechanisms.

Theorem (Lavi, Mu'alem and Nisan, FOCS 2003)

For the combinatorial auction problem the generalized VCG is essentially the only truthful mechanism, under some mild (?) assumptions.

Our characterization

Theorem

For the scheduling unrelated machines problem, the decisive truthful mechanisms for 2 tasks and real values are either task-independent or affine minimizers.
(The same characterization applies to mechanisms that are decisive for only three outcomes.)

Theorem (Generalization for many tasks)

A decisive truthful mechanism partitions the tasks into groups so that the tasks in each group are allocated independently of the other groups. Tasks in a group of size at least two are allocated by an affine minimizer and tasks in singleton groups by a task-independent mechanism.

Our characterization

Theorem

For the scheduling unrelated machines problem, the decisive truthful mechanisms for 2 tasks and real values are either task-independent or affine minimizers.
(The same characterization applies to mechanisms that are decisive for only three outcomes.)

Theorem (Generalization for many tasks)

A decisive truthful mechanism partitions the tasks into groups so that the tasks in each group are allocated independently of the other groups. Tasks in a group of size at least two are allocated by an affine minimizer and tasks in singleton groups by a task-independent mechanism.

Conjecture

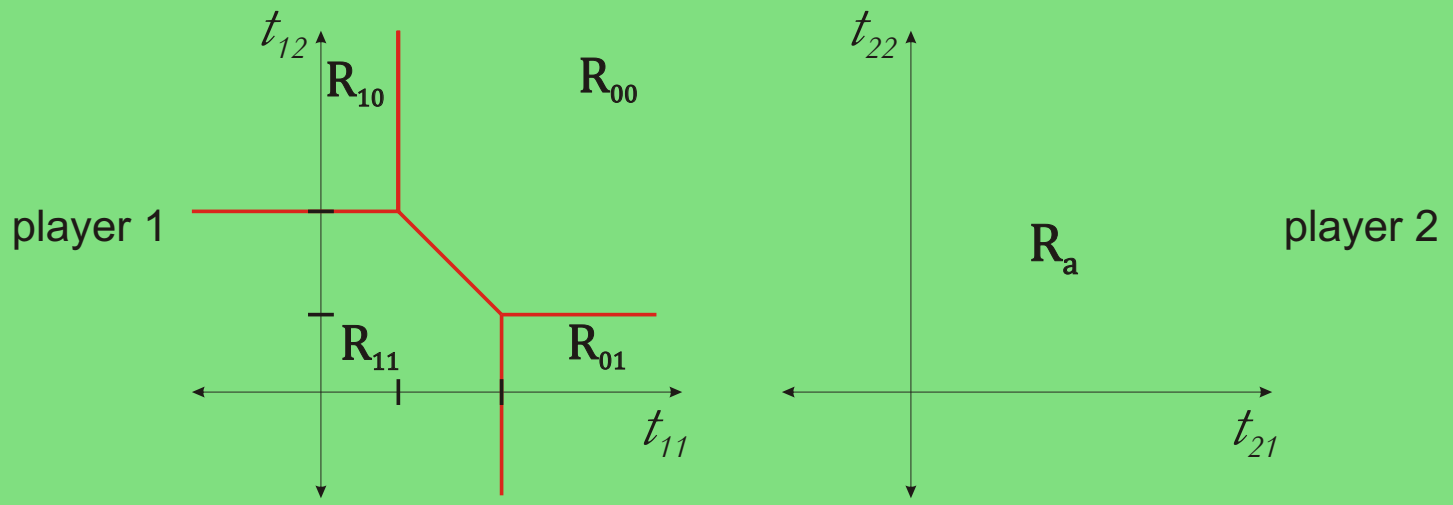
For any number of players, a decisive truthful mechanism partitions the tasks into groups. Each group of tasks is allocated by either an affine minimizer or a threshold mechanism.

What if we raise decisiveness?

A mechanism that is neither affine minimizer nor threshold

Mechanism with some oblivious player

Suppose the allocation is given by $\text{argmin}_a\{a_{11}t_{11} + a_{12}t_{12} + \gamma_a\}$ where $\gamma_{11} = 0$ and $\gamma_{10} = \gamma_{01} = 2, \gamma_{00} = 3$.



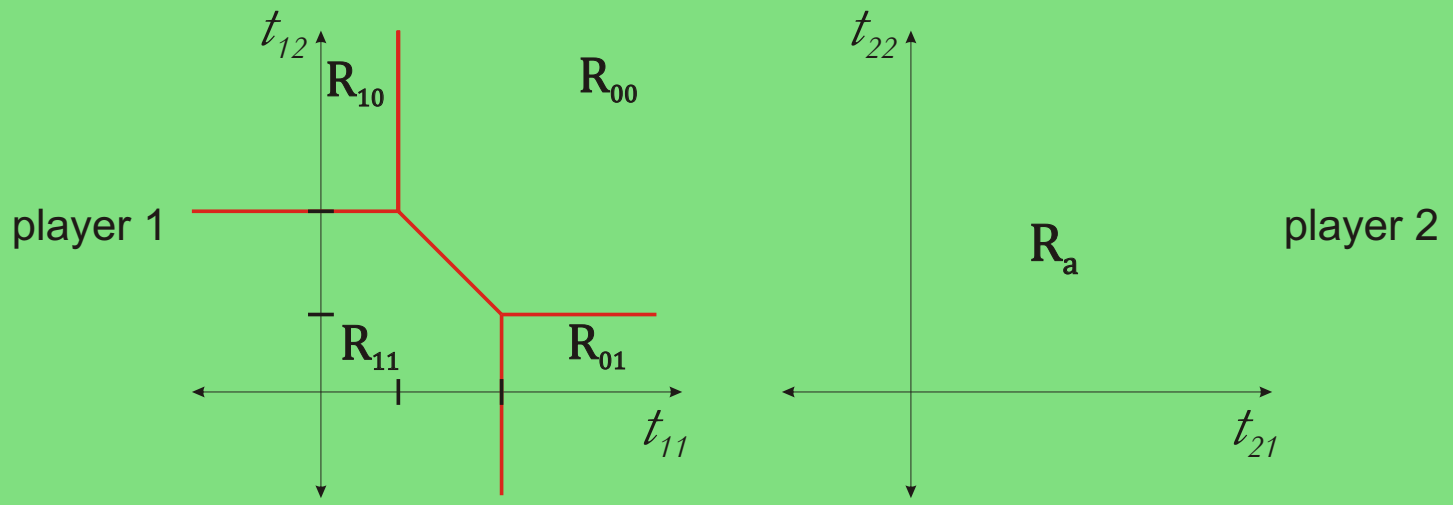
player 1 is decisive: he has all regions for any fixed t_2,
player 2 oblivious: his values do not affect the allocation so he has only one region for fixed t_1.

What if we raise decisiveness?

A mechanism that is neither affine minimizer nor threshold

Mechanism with some oblivious player

Suppose the allocation is given by $\text{argmin}_a\{a_{11}t_{11} + a_{12}t_{12} + \gamma_a\}$ where $\gamma_{11} = 0$ and $\gamma_{10} = \gamma_{01} = 2, \gamma_{00} = 3$.



player 1 is decisive: he has all regions for any fixed t_2,
player 2 oblivious: his values do not affect the allocation so he has only one region for fixed t_1.

By taking values in $(-\infty, \infty)$ instead of $(0, \infty)$ we make the decisiveness assumption milder.

Minimizing the Makespan for 2 players

Applying the characterization to reason about the approximation ratio.

Theorem (Dobzinski and Sundararajan EC'08)

Any mechanism with bounded approximation ratio is task-independent.

(we give a very simple proof)

Theorem

No mechanism for 2 (or more) tasks has approximation ration< 2. The VCG is the unique mechanism that achieves approx. ratio$= 2$.

The existing lower bound was for 3 (or more) tasks.

$$\begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix} \qquad \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix}$$

Open Problems

- The major open problem is to bridge the gap between the lower bound of 2.61 and the upper bound of n (and the same problem for the fractional and randomized mechanisms).
- But generalizing this characterization for more players would be even more important and would also give a result directly applicable to the combinatorial auctions domain.