SECOND PART on Non-cooperative Networks: Algorithmic Mechanism Design

Game Theory vs Mechanism Design (in poor words)

Game theory aims to investigate rational decision-making in conflicting situations, whereas mechanism design just concerns the reverse question: given some desirable outcome, can we design a game that produces it (in equilibrium)?

The implementation problem (informally)

Imagine that you are a planner who develops criteria for social welfare, and you want to design a game such that the equilibrium of the game conforms to some concept of social optimality (i.e., aggregation of players' preferences w.r.t. to a certain outcome). However, these preferences of individuals are now a private information, that you have to disclose in order to precisely evaluate the social utility of a certain outcome. Observe that it may be in the best interest of some player to lie about her preference: Indeed, this may lead to a certain outcome which improves her personal benefit, regardless if this may negatively affect other players! Thus, in this strategic setting, which techniques can be used to convince players to cooperate honestly with the system by revealing the truth???

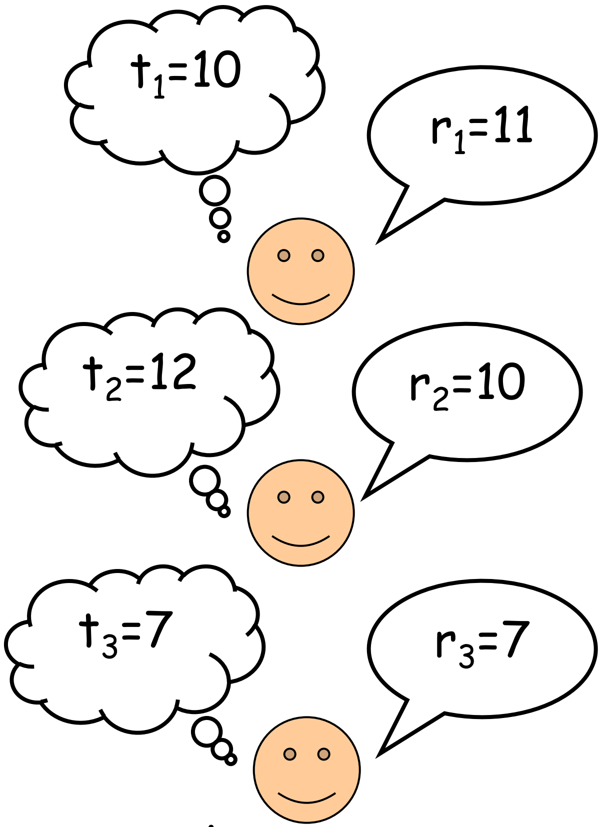
Designing a Mechanism

- Informally, designing a mechanism means to define a game in which a desired outcome must be reached (in equilibrium)
- However, games induced by mechanisms are different from games seen so far:
  - Players hold independent **private values**, called types
  - The payoffs are a function of these types

  $\Rightarrow$ each player does not really know about the other players' payoffs, but only about her one!

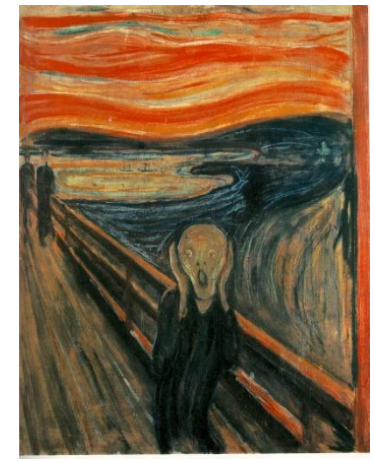
$\Rightarrow$ Games with incomplete information

An example: sealed-bid auctions



SCF: the winner should be the guy **having in mind** the highest value for the painting

r_i: is the amount of money player i **bids** (in a sealed envelope) for the painting



t_i: is the **maximum** amount of money player i is willing to pay for the painting, i.e., her **valuation** of the painting in case she will get it

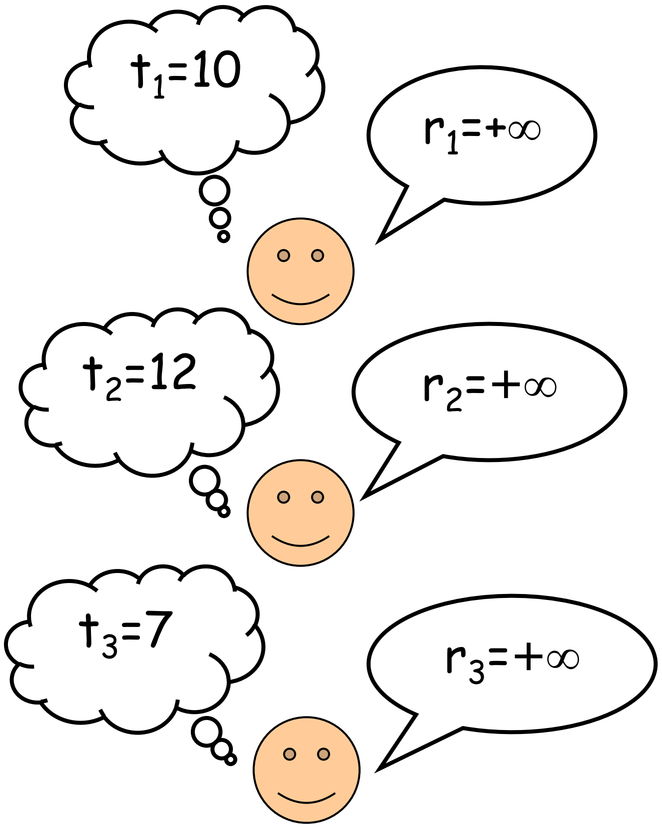
If player i wins and has to pay p then her **utility** is $u_i=t_i-p$, otherwise it is 0

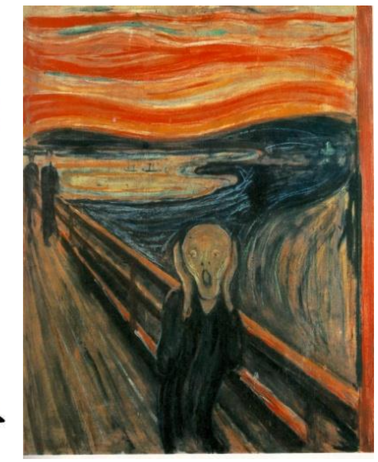
The mechanism tells to players:

(1) How the item will be allocated (i.e., who will be the winner), depending on the received bids

(2) The payment the winner has to return, as a function of the received bids

A simple mechanism: no payment

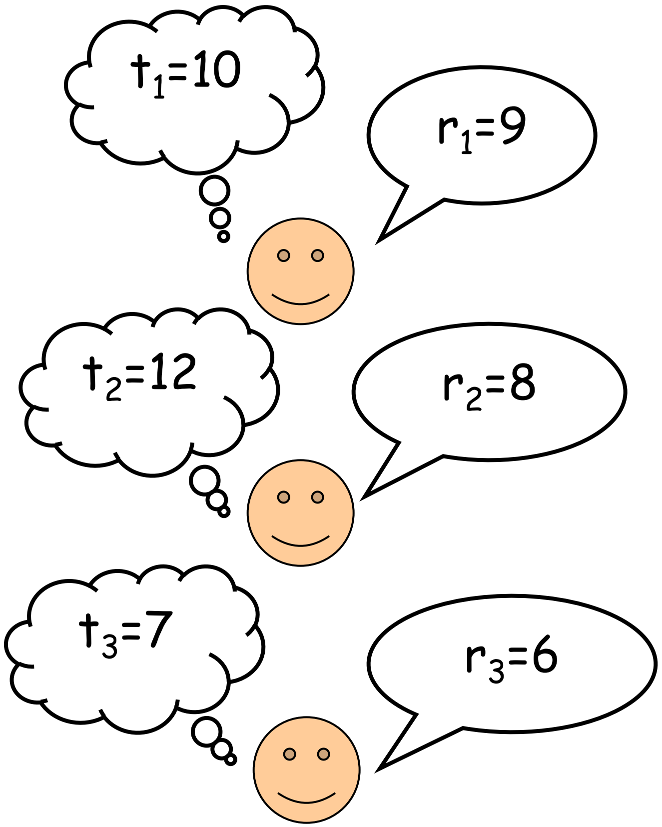




Mechanism: The highest bid wins and the price of the item is 0

…it doesn't work…

Another simple mechanism: pay your bid



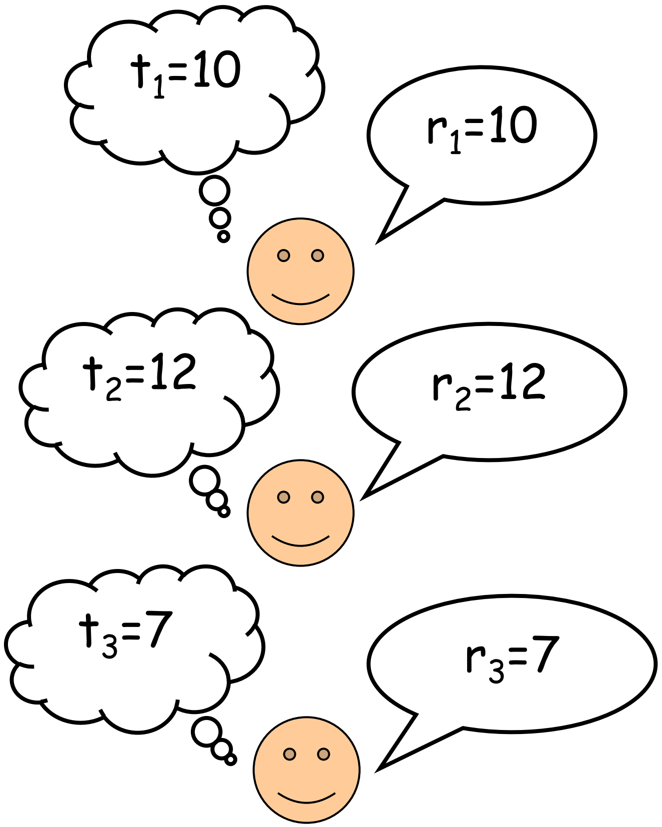
Mechanism: The highest bid wins and the winner will pay her bid

Player i may bid $r_i < t_i$ (in this way she is guaranteed not to incur a negative utility)

…and so the winner could be the wrong one…

…it doesn't work…

An elegant solution: Vickrey's second price auction



Mechanism: The highest bid wins and the winner will pay the second highest bid

every player has convenience to declare the truth!
(we prove it in the next slide)

Theorem

In the Vickrey auction, for every player i, $r_i = t_i$ is a dominant strategy

proof Fix i and t_i, and look at strategies for player i. Let $R = \max_{j \neq i} \{r_j\}$.

Case $t_i > R$ (observe that R is unknown to player i)

1. declaring $r_i = t_i$ gives utility $u_i = t_i - R > 0$ (player wins)
2. declaring any $r_i > R$, $r_i \neq t_i$, yields again utility $u_i = t_i - R > 0$ (player wins)
3. declaring $r_i = R$ yields a utility depending on the tie-breaking rule: if player i wins, she has again utility $u_i = t_i - R > 0$, while if she loses, then $u_i = 0$
4. declaring any $r_i < R$ yields $u_i = 0$ (player loses)

$\Rightarrow$ In any case, the best utility is $u_i = t_i - R$, which is obtained when declaring $r_i = t_i$

Case $t_i < R$

1. declaring $r_i = t_i$ yields utility $u_i = 0$ (player loses)
2. declaring any $r_i < R$, $r_i \neq t_i$, yields again utility $u_i = 0$ (player loses)
3. declaring $r_i = R$ yields a utility depending on the tie-breaking rule: if player i wins, she has utility $u_i = t_i - R < 0$, while if she loses, then she has again utility $u_i = 0$
4. declaring any $r_i > R$ yields $u_i = t_i - R < 0$ (player wins)

$\Rightarrow$ In any case, the best utility is $u_i = 0$, which is obtained when declaring $r_i = t_i$

Proof (cont'd)

Case t_i = $\mathbf{R}$

1. declaring r_i=t_i yields utility u_i = t_i-$\mathbf{R}$ = 0 (player wins/loses depending on the tie-breaking rule, but her utility in this case is always 0)
2. declaring any r_i < $\mathbf{R}$ yields again utility u_i= 0 (player loses)
3. declaring any r_i > $\mathbf{R}$ yields u_i= t_i-$\mathbf{R}$ = 0 (player wins)

$\Rightarrow$ In any case, the best utility is u_i= 0, which is obtained when declaring r_i=t_i

$\Rightarrow$ In all the cases, reporting a false type produces a not better utility, and so telling the truth is a dominant strategy! ■

Mechanism Design Problem: ingredients

- N players; each player i, i=1,..,N, has some **private** information $t_i \in T_i$ (actually, this is the **only** private information of the game, all the other functions provided in the following are public) called **type**
  - Vickrey's auction: the type is the value of the painting that a player has in mind, and so T_i is the set of positive real numbers
- A set of **feasible outcomes** X (i.e., the result of the interaction of the players with the mechanism)
  - Vickrey's auction: X is the set of players (indeed an outcome of the auction is a winner of it, i.e., a player)

Mechanism Design Problem: ingredients (2)

- For each vector of types $t=(t_1, t_2, \ldots, t_N)$, and for each feasible outcome $x \in X$, a SCF $f(t,x)$ that measures the quality of x as a function of t. This is the function that the mechanism aims to implement (i.e., it aims to select an outcome x^* that minimizes/maximizes it, but the problem is that types are unknown!)
  - Vickrey's auction: $f(t,x)$ is the type associated with a feasible winner x (i.e., any of the players), and the objective is to maximize f, i.e., to allocate the painting to the bidder with highest type

- Each player has a **strategy space** S_i and performs a strategic action; we restrict ourselves to *direct revelation mechanisms*, in which the action is reporting a value r_i from the type space (with possibly $r_i \neq t_i$), i.e., $S_i = T_i$
  - Vickrey's auction: the action is to bid a value r_i

Mechanism Design Problem: ingredients (3)

- For each feasible outcome $x \in X$, each player makes a **valuation** $v_i(t_i,x)$ (in terms of some common currency), expressing her preference about that output x
  - Vickrey's auction: if player i wins the auction then her valuation is equal to her type t_i, otherwise it is 0
- For each feasible outcome $x \in X$, each player receives a **payment** $p_i(x)$ by the system in terms of the common currency (a negative payment means that the player makes a payment to the system); payments are used by the system to incentive players to be collaborative.
  - Vickrey's auction: if player i wins the auction then she "receives" a payment equal to $-r_j$, where r_j is the second highest bid, otherwise it is 0
- Then, for each feasible outcome $x \in X$, the **utility** of player i (in terms of the common currency) coming from outcome x will be:

$$u_i(t_i,x) = p_i(x) + v_i(t_i,x)$$

  - Vickrey's auction: if player i wins the auction then her utility is equal to $u_i = -r_j+t_i \geq 0$, where r_j is the second highest bid, otherwise it is $u_i = 0+0=0$

Our focus: Truthful (or Strategy-proof) Mechanism Design

Given all the above ingredients, **design** a **mechanism** $M=<g, p>$, where:

- $g:S_1\times\ldots\times S_N \rightarrow X$ is an **algorithm** which computes an outcome $g(r)\in X$ as a function of the reported types r
- $p(g(r))=(p_1(g(r)),\ldots,p_N(g(r)))\in\Re^N$ is a **payment scheme** w.r.t. outcome $g(r)$ that specifies a payment for each player

which **implements** (i.e., optimize) the SCF $f(t,x)$ in **dominant strategy equilibrium** w.r.t. players' utilities whenever players report their **true types**. Such a mechanism is called a **truthful** (or **strategy-proof**) **mechanism**.

(In other words, with the reported type vector $r=t$ the mechanism provides a solution $g(t)$ and a payment scheme $p(g(t))$ such that players' utilities $u_i(t_i,g(t)) = p_i(g(t)) + v_i(t_i,g(t))$ are maximed in DSE and $f(t,g(t))$ is optimal (either minimum or maximum)).

Truthful Mechanism Design in DSE: **Economics Issues**

QUESTION: How to design a truthful mechanism? Or, in other words:

1. How to design the algorithm g, and
2. How to define the payment scheme p

in such a way that the underlying SCF is implemented truthfully in DSE? Under which conditions can this be done?

Algorithmic Mechanism Design

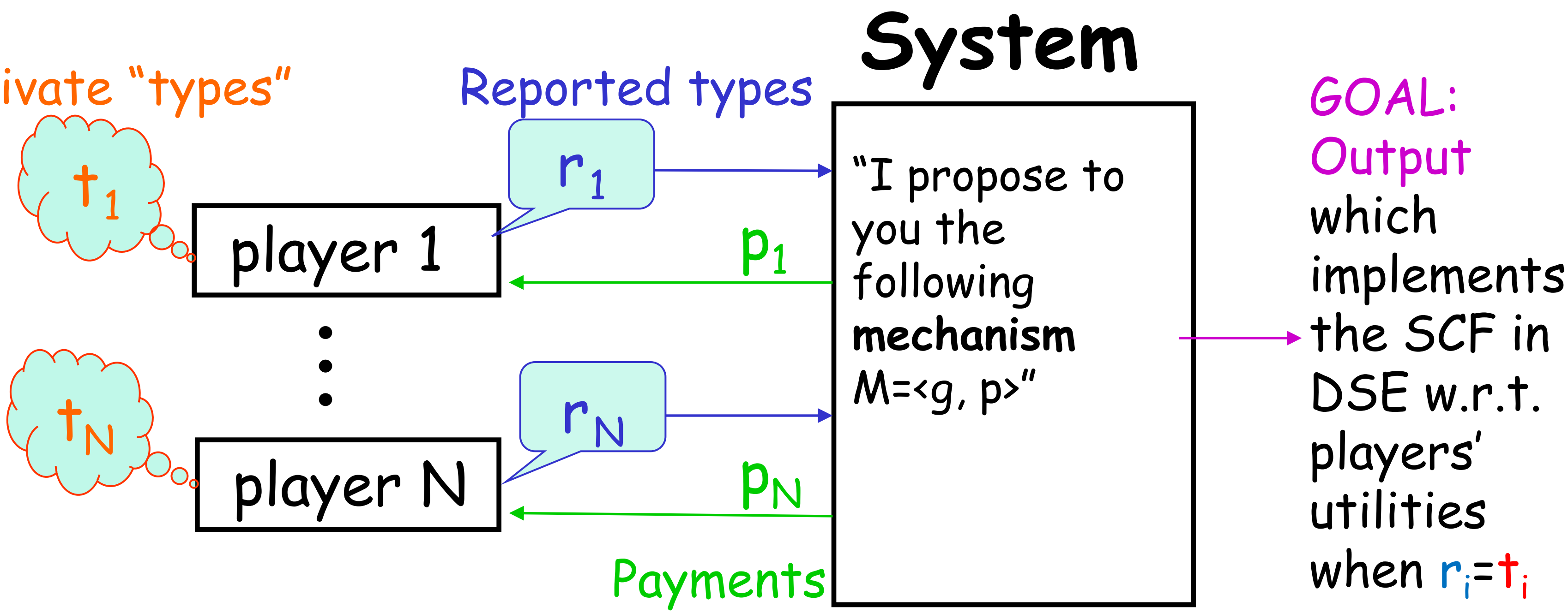
QUESTION: What is the time complexity of the mechanism? Or, in other words:

- What is the time complexity of computing g(r)?
- What is the time complexity to calculate the N payment functions?
- What does it happen if it is **NP-hard** to implement the underlying SCF?

Question: What is the time complexity of the Vickrey auction?

Answer: $\Theta(N)$, where N is the number of players. Indeed, it suffices to check all the offers, by keeping track of the largest one and of the second largest one.

Mechanism Design: a picture



Each player reports strategically to maximize her well-being…

…in response to a payment which is a function of the output!

A prominent class of problems

- **Utilitarian problems:** A problem is *utilitarian* if its SCF is such that $f(t,x) = \sum_i v_i(t_i,x)$, i.e., the SCF is separately-additive w.r.t. players' valuations.

Remark 1: the auction problem is utilitarian, in that $f(t,x)$ is the type associated with the winner x, and the valuation of a player is either her type or 0, depending on whether she wins or not. Then, $f(t,x) = \sum_i v_i(t_i,x)$ = type of the winner

Remark 2: in many network optimization problems (which are of our special interest) the SCF is separately-additive

Good news: for utilitarian problems there exists a class of truthful mechanisms ☺

Vickrey-Clarke-Groves (VCG) Mechanisms

- A VCG-mechanism is (the only) strategy-proof mechanism for **utilitarian** problems:
  - Algorithm g computes:

$$g(r) = \arg\max_{y \in X} \sum_i v_i(r_i, y)$$

  - Payment function for player i:

$$p_i(g(r)) = h_i(r_{-i}) + \sum_{j \neq i} v_j(r_j, g(r))$$

  where $h_i(r_{-i}) = h(r_1, r_2, \ldots, r_{i-1}, r_{i+1}, \ldots, r_N)$ is an arbitrary function of the types reported by players other than player i.
- What about **non-utilitarian** problems? Strategy-proof mechanisms are known only when the type is a single parameter.

Theorem

VCG-mechanisms are truthful for utilitarian problems

Proof: We show that a player has no interest in lying.

Fix i, r_{-i}, t_i. Let $\check{r}=(r_{-i},t_i)$ and consider a strategy $r_i \neq t_i$

$x=g(r_{-i},t_i)=g(\check{r})$ $\quad$ $x'=g(r_{-i},r_i)$ $\quad$ $\check{r}_j=r_j$ if $j\neq i$, and $\check{r}_i=t_i$

$$u_i(t_i,x) = [h_i(r_{-i}) + \Sigma_{j\neq i} v_j(r_j,x)] + v_i(t_i,x) = h_i(r_{-i}) + \Sigma_j v_j(\check{r}_j,x)$$

$$u_i(t_i,x') = [h_i(r_{-i}) + \Sigma_{j\neq i} v_j(r_j,x')] + v_i(t_i,x') = h_i(r_{-i}) + \Sigma_j v_j(\check{r}_j,x')$$

but x is an optimal solution w.r.t. $\check{r}=(r_{-i},t_i)$, i.e.,

$$x = \arg\max_{y\in X} \sum_j v_j(\check{r}_j,y)$$

$\Rightarrow$ $\Sigma_j v_j(\check{r}_j,x) \geq \Sigma_j v_j(\check{r}_j,x')$ $\Rightarrow$ $u_i(t_i,x) \geq u_i(t_i,x')$.

■

How to define $h_i(r_{-i})$?

Remark: not all functions make sense. For instance, what does it happen in our Vickrey's auction if we set for every player $h_i(r_{-i})=-1000$ (notice this is independent of reported value r_i of player i, and so it obeys to the definition)?

Answer: It happens that players' utility become negative; more precisely, the winner's utility is

$u_i(t_i,x) = p_i(x) + v_i(t_i,x) = h_i(r_{-i}) + \sum_{j\neq i} v_j(r_j,x) + v_i(t_i,x) = -1000+0+12 = -988$

while utility of losers is

$u_i(t_i,x) = p_i(x) + v_i(t_i,x) = h_i(r_{-i}) + \sum_{j\neq i} v_j(r_j,x) + v_i(t_i,x) = -1000+12+0 = -988$

$\Rightarrow$This is undesirable in reality, since with such perspective players would not participate to the auction!

Voluntary participation

A mechanism satisfies the voluntary participation condition if players who reports truthfully never incur a net loss, i.e., for all players i, true values t_i, and other players' bids r_{-i}

$$u_i(t_i, g(r_{-i}, t_i)) \geq 0.$$

The Clarke payments

solution maximizing the sum of valuations when player i doesn't play

- This is a special VCG-mechanism in which

$$h_i(r_{-i}) = -\sum_{j\neq i} v_j(r_j, g(r_{-i}))$$

$$\Rightarrow \quad p_i(g(r)) = -\sum_{j\neq i} v_j(r_j, g(r_{-i})) + \sum_{j\neq i} v_j(r_j, g(r))$$

- With Clarke payments, it can be shown that players' utility are always non-negative; indeed:

$$u_i(t_i, g(r)) = p_i(g(r)) + v_i(t_i, g(r)) = -\sum_{j\neq i} v_j(r_j, g(r_{-i})) + \sum_{j\neq i} v_j(r_j, g(r)) + v_i(t_i, g(r)) = -\sum_{j\neq i} v_j(r_j, g(r_{-i})) + \sum_{j} v_j(r_j, g(r)) \geq 0$$

since the first term is never larger (in absolute value) than the second one (intuitively, adding one more player will never decrease the social welfare)

$\Rightarrow$ players are interested in playing the game

The Vickrey's auction is a VCG mechanism with Clarke payments

- Recall that auctions are **utilitarian** problems. Then, the VCG-mechanism associated with the Vickrey's auction is:
  - $g(r) = \arg\max_{y \in X} \sum_i v_i(r_i, y)$

    …this is equivalent to allocate to the bidder with **highest reported cost** (in the end, the **highest type**, since it is strategy-proof)
  - $p_i(g(r)) = -\sum_{j \neq i} v_j(r_j, g(r_{-i})) + \sum_{j \neq i} v_j(r_j, g(r))$

    …this is equivalent to say that the winner pays the **second highest reported cost** (in the end, the **second highest type**, since it is strategy-proof), and the losers pay 0, respectively

Remark: the difference between the second highest offer and the highest offer is unbounded (frugality issue)

VCG-Mechanisms: Advantages

- For System Designer:
  - The goal, i.e., the optimization of the SCF, is achieved with certainty
- For players:
  - players have truth telling as the dominant strategy, so they need not require any computational systems to deliberate about other players strategies

VCG-Mechanisms: Disadvantages

- For System Designer:
  - The payments may be sub-optimal (frugality)
  - Apparently, with Clarke payments, the system may need to run the mechanism's algorithm N+1 times: once with all players (for computing the outcome $g(r)$), and once for every player (indeed, for computing the payment p_i associated with player i, we need to know $g(r_{-i})$)

  $\Rightarrow$ If the problem is hard to solve then the computational cost may be very heavy
- For players:
  - players may not like to tell the truth to the system designer as it can be used in other ways

Algorithmic mechanism design and network protocols

- Large networks (e.g., Internet) are built and controlled by diverse and competitive entities:
  - Entities own different components of the network and hold private information
  - Entities are selfish and have different preferences

$\Rightarrow$ Mechanism design is a useful tool to design protocols working in such an environment, but time complexity is an important issue due to the massive network size

Algorithmic mechanism design for network optimization problems

- Simplifying the Internet model, we assume that each player owns a **single edge** of a graph $G=(V,E)$, and privately knows the **cost** for using it

$\Rightarrow$ Classic optimization problems on G become **private-edge mechanism design optimization problems**, in which the player's type is the **weight** of the edge!

- Many basic network design problems have been studied in this framework: shortest path (SP), single-source shortest-path tree (SPT), minimum spanning tree (MST), and many others
- Remark: Quite naturally, SP and MST are utilitarian problems: indeed the cost of a solution (social-choice function) is simply the sum of the edge costs
- On the other hand, the SPT is not! Can you see why?

Some remarks

- In general, network optimization problems are minimization problems (the Vickrey's auction was instead a maximization problem)
- Accordingly, we have:
  - for each $x \in X$, the valuation function $v_i(t_i,x)$ represents a cost incurred by player i in the solution x (and so it is a negative function of its type)
  - the social-choice function $f(t,x)$ is negative (since it is an "aggregation" of negative valuation functions), and so its maximization corresponds to a minimization of the costs incurred by the players
  - payments are now from the mechanism to players (i.e., they are positive)

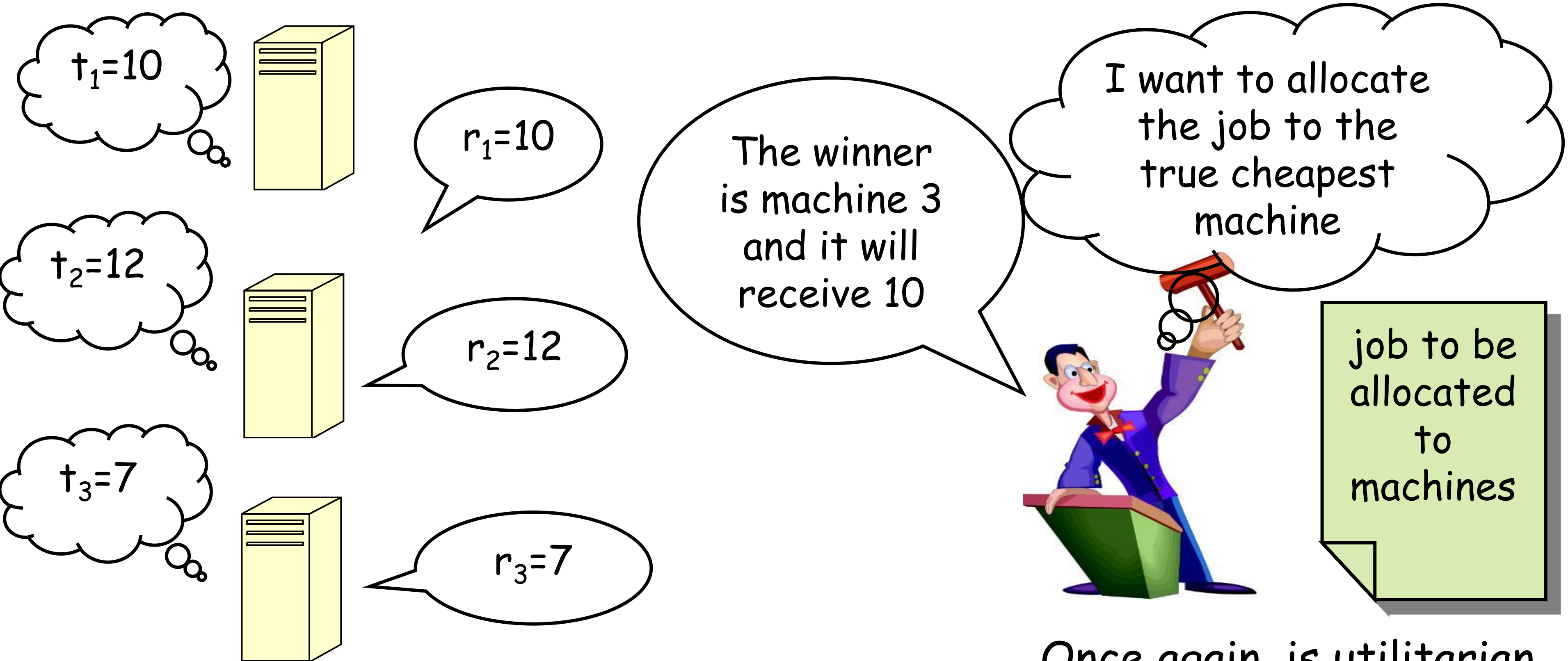
Summary of main results

	Centralized algorithm	Private-edge mechanism
SP	$O(m+n \log n)$	$O(m+n \log n)$ (VCG)
MST	$O(m\ \alpha(m,n))$*	$O(m\ \alpha(m,n))$ (VCG)
SPT	$O(m+n \log n)$	$O(m+n \log n)$ (single-parameter)

$\alpha(m,n)$ is the extremely slow-growing inverse of the Ackermann function

$\Rightarrow$ For all these basic problems, the time complexity of the mechanism equals that of the canonical centralized algorithm!

Exercise: redefine the Vickrey auction in the minimization version (so-called **procurement auction**)



t_i: cost incurred by i if he does the job
v_i: is equal to $-t_i$ if i is the winner, and 0 otherwise
p_i: is equal to the second highest type if i is the winner, and 0 otherwise

Once again, is utilitarian, and so the the second price auction (VCG mechanism) is truthful: the cheapest bid wins and the winner will get the second cheapest bid