

# Game Theory

## Auction design 3: VCG

Xiang Sun

2021 Summer

# Multiple-item auction

- Auctions are often about selling **multiple items** at the same time: timber, spectrum, oil fields, etc.
- But it is often the case that the buyers want only a **combination** of the items (a subset of the items). However,
  - bidders may want to participate in more auctions than the number of items they want;
  - bidders do not want to buy more items than needed.
- Bidders are not obliged to buy several items: 1 item only is also a combination.
- The **Vickrey-Clarke-Groves (VCG)** auction is the solution to this problem.

# The model

- $B = \{1, 2, \dots, n\}$  is the set of bidders.
- $X = \{x_1, x_2, \dots, x_k\}$  is the set of items to be sold.
- Each bidder  $i$  has a valuation  $v_i$  over the **sets** of items.
- Suppose that bidder 1's most preferred is to have items  $x_1$  and  $x_2$ , and then  $x_2$  and  $x_3$ , and then  $x_1, x_2, x_3$ . His/her valuation would be then (for instance):
  - $v_1(\{x_1, x_2\}) = 20$ ;
  - $v_1(\{x_2, x_3\}) = 15$ ;
  - $v_1(\{x_1, x_2, x_3\}) = 10$ .

# Valuation function

- More generally, each bidder has a **valuation function** that assigns to each **set of items** a value.
- So bidder  $i$ 's valuation is a function  $v_i$  that assigns for each subset  $S \subseteq X$  a valuation  $v_i(S)$ :

$$v_i: S \mapsto v_i(S).$$

- Interpretation:  $v_i(S)$  is how much bidder  $i$  is willing to pay to get the items  $S$ :
  - all the items that are contained in  $S$ ;
  - no item that is not contained in  $S$ .

# VCG auction

The VCG auction works as follows:

- 1 Each bidder submits his **bid function**.  
⇒ it is like a sealed-bid auction.
- 2 The assignment (who gets what) is the one that **maximizes the social value**.
- 3 Each bidder pays the **externality** she imposes on the other bidders.

# Maximizing the social value

- Suppose we are the auctioneer and we already collected each bidder's bid functions.
- The first objective is to find an **assignment** of the items (i.e., which item goes to whom)  $\mu: N \rightarrow 2^X$ ,

$\mu(i)$  = the set of items that goes to bidder  $i$ .

- The assignment  $\mu$  must be such that **no two bidders get assigned the same item**.

If  $x \in \mu(i)$  (i.e., bidder  $i$  gets item  $x$ ), then there is no bidder  $j \neq i$  such that  $x \in \mu(j)$ .

## Maximizing the social value (Cont.)

The assignment of interest is the assignment  $\mu^*$  that maximizes the **social value**:

$$\sum_{i \in N} v_i(\mu^*(i)) \geq \sum_{i \in N} v_i(\mu(i)) \text{ for any other assignment } \mu,$$

where for each bidder  $i$ ,  $v_i$  is the bid function submitted by  $i$ .

- $\mu^*(i)$  is the assignment of bidder  $i \rightarrow$  a set of items.
- $v_i(\mu^*(i))$  is how much bidder  $i$  claimed she values the set of items  $\mu^*(i)$ .

# Why maximizing the social value?

Take any auction,

- $v$  is the value of the buyer.
- $p$  is the price.
- social surplus =  $\underbrace{p}_{\text{seller's surplus}} + \underbrace{(v - p)}_{\text{winner's surplus}} + \underbrace{(n - 1) \times 0}_{\text{losers' surplus}} = v.$

For standard auctions (2nd price, Dutch, etc.) and environments (private and independent values):

- Winner = bidder with the highest valuation.  
 $\Rightarrow$  social surplus is maximized.
- VCG follows the same principle: to implement the assignment that maximizes social value/surplus.



# Paying the externality

Once we have the optimal assignment, we do for each bidder  $i$  the following:

- Consider the set of all bidders **except bidder  $i$** :  $N \setminus \{i\}$ .
- Find the assignment  $\mu$  of the items in  $X$  that maximizes the social value **over the set  $N \setminus \{i\}$** :

$$\max \sum_{j \neq i} v_j(\mu(j)).$$

- \* The assignment that maximizes this sum is the one that would prevail if bidder  $i$  were not present.
- The price that bidder  $i$  has then to pay for the items in  $\mu^*(i)$  is then

$$\underbrace{\max \sum_{j \neq i} v_j(\mu(j))}_{\text{others' social value without bidder } i} - \underbrace{\sum_{j \neq i} v_j(\mu^*(j))}_{\text{others' social value with bidder } i}.$$

## Example

Suppose there are two bidders, 1 and 2, and two items,  $x_1$  and  $x_2$ . Assume that the valuations are

	Bidder 1	Bidder 2
$x_1$	10	5
$x_2$	5	3
$x_1$ & $x_2$	11	6

For each possible assignment we compute the social value:

	Bidder 1	Bidder 2	social value
assignment 1	$x_1, x_2$		11
assignment 2		$x_1, x_2$	6
assignment 3	$x_1$	$x_2$	13
assignment 4	$x_2$	$x_1$	10

## Example (Cont.)

The assignment that maximizes the social value is then assignment 3, so we have

$$\mu^*(1) = x_1, \mu^*(2) = x_2.$$

- If bidder 1 is absent, bidder 2 can get both items and her valuation is 6.
- When bidder 1 is present bidder 2 gets only  $x_2$ , with a valuation of 3.
- So bidder 1's price for  $x_1$  is  $6 - 3 = 3$ .

## Example (Cont.)

The assignment that maximizes the social value is then assignment 3, so we have

$$\mu^*(1) = x_1, \mu^*(2) = x_2.$$

- If bidder 2 is absent, bidder 1 can get both items and her valuation is 11.
- When bidder 2 is present bidder 1 gets only  $x_1$ , with a valuation of 10.
- So bidder 2's price for  $x_2$  is  $11 - 10 = 1$ .

# VCG auction vs. Vickrey auction

When there is only one item to be sold, the VCG auction is simply the Vickrey auction:

- $i$  = bidder with the highest valuation.
- $j$  = bidder with the second highest valuation.
- If  $i$  is here: assignment that maximizes the social value:
  - $i$  gets the item.
  - $j$  gets nothing.
  - All other bidders get nothing.
- If  $i$  is not here: assignment that maximizes the social value:
  - $i$  does **not** get the item (she's not here).
  - $j$  gets the item.
  - All other bidders get nothing.

## VCG auction vs. Vickrey auction (Cont.)

Let  $v_j$  = valuation of bidder  $j$ .

$$\begin{aligned}\text{price for bidder } i &= \text{social value of others if } i \text{ not here} \\ &\quad - \text{social value of others if } i \text{ here} \\ &= (v_j + 0 + \cdots + 0) - (0 + 0 + \cdots + 0) \\ &= v_j = \text{2nd highest valuation.}\end{aligned}$$

$\Rightarrow$  The winner pays the second highest valuation.

### Remark

The VCG **action** presented here is a special case of a more general object, the VCG **mechanism**. To make the distinction some authors called this VCG auction the **generalized Vickrey auction**.

# Truthful bidding under VCG

- The auctioneer does not know the bidders' valuations. She only has the bids: everything is calculated using the bids.
- Bidders may not be forced to submit their true valuation functions.
- It turns out that submitting one's **true** valuation function is a **dominant strategy** in the VCG auction.

# Proof

Suppose that bidder  $i$  bids  $\hat{v}_i(\cdot)$  instead of her true valuation,  $v_i(\cdot)$ .

Define:

- $\hat{\mu}$  is the assignment that maximizes  $(v_1, \dots, v_{i-1}, \hat{v}_i, v_{i+1}, \dots, v_n)$ .
- $\hat{V}_{N \setminus \{i\}} = \sum_{j \neq i} v_j(\hat{\mu}(j))$ .
- $V_{N \setminus \{i\}} = \max_{\mu} \sum_{j \neq i} v_j(\mu(j)) \leftarrow$  Does not depend on  $i$ 's bid.
- $V_{N \setminus \{i\}}^* = \sum_{j \neq i} v_j(\mu^*(j))$ .

Let

$$\begin{aligned} U_i &= \text{net payoff when bidding truthfully } v_i(\cdot) \\ &= v_i(\mu^*(i)) - (V_{N \setminus \{i\}} - V_{N \setminus \{i\}}^*), \end{aligned}$$

and

$$\begin{aligned} \hat{U}_i &= \text{net payoff when bidding } \hat{v}_i(\cdot) \\ &= v_i(\hat{\mu}(i)) - (V_{N \setminus \{i\}} - \hat{V}_{N \setminus \{i\}}). \end{aligned}$$



## Proof (Cont.)

Compute the difference

$$\begin{aligned} U_i - \hat{U}_i &= v_i(\mu^*(i)) - (V_{N \setminus \{i\}} - V_{N \setminus \{i\}}^*) \\ &\quad - (v_i(\hat{\mu}(i)) - (V_{N \setminus \{i\}} - \hat{V}_{N \setminus \{i\}})) \\ &= \underbrace{(v_i(\mu^*(i)) + V_{N \setminus \{i\}}^*)}_{\sum_j v_j(\mu^*(j))} - \underbrace{(v_i(\hat{\mu}(i)) + \hat{V}_{N \setminus \{i\}})}_{\sum_j v_j(\hat{\mu}(j))}. \end{aligned}$$

Recall that  $\mu^*$  is the assignment that maximizes the social value. So,

$$\sum_j v_j(\mu^*(j)) - \sum_j v_j(\hat{\mu}(j)) \iff U_i - \hat{U}_i \geq 0.$$

That is, bidder  $i$ 's highest net payoff is higher when bidding truthfully.

# Problem with the VCG auction

In practice, the VCG auction is not used in many cases. Why?

- A crucial step is to find an assignment that maximizes the social welfare.
- Amounts to compare all possible assignments: this can be a **gigantic** number.
- We also need to ask bidders to submit a valuation function over a **large** number of possibilities.

If there are 10 objects, the bidder would have to submit  $2^{10} = 1024$  different valuations.

## Example

- 10 bidders, for 10 items: each bidder only want 1 item.
- To find the optimal assignment  $\mu^*$ :  $10! = 3,628,800$  combinations.
- For each bidder, the optimal assignment when not present:  $10! = 3,628,800$  combinations.
- So to compute payoffs one need:  $10! + 10 \times 10! = 39,916,800$ .

Computer scientists say that this problem is “NP complete”:

- We don't know how to find a solution in “reasonable time”.
- If we multiply by 2 the size of the problem (number of bidders and objects) the number of calculations increases by a factor more than 2 (exponential).

# Take-away

- **Combinatorial auctions** address the problem of selling multiple items at the same time.
- Bidder's valuations in such settings specify a value **for each** combination of items.
- In the Vickrey-Clarke-Groves auction we first assign the items so that the **social valuation** is the highest (the **optimal valuation**).
- Each bidder pays the **externality** she imposes on the others:

$$\begin{aligned} \text{price} &= \text{others' max social value without the bidder} \\ &\quad - \text{others' social value at optimal assignment.} \end{aligned}$$

## Take-away (Cont.)

- The VCG auction **generalizes** the Vickrey (second-price) auction. With one item only they are identical auction formats.
- The VCG auction is **strategy-proof**: it is a dominant strategy to bid one's true valuations.
- VCG may be difficult to use in practice if there are too many buyers and items: the number of assignments grows exponentially.