

## VCG Mechanisms

Thomas Kesselheim

Last Update: May 25, 2025

So far, we considered single-parameter environments for mechanism-design problems. We found a characterization of truthful mechanisms, making it somewhat easy to design a truthful mechanism. For general settings, such a nice characterization does not exist. However, if the task is to maximize social welfare and we do not care too much about computational issues, there is a very elegant solution due to Vickrey, Clarke, and Groves.

## 1 Motivating Example: Combinatorial Auction

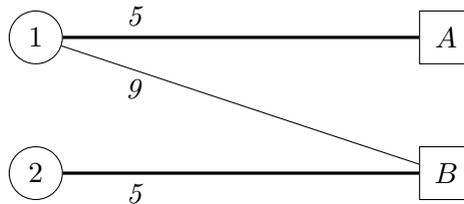
In a combinatorial auction, we have a set  $N$  of  $n$  bidders and a set  $M$  of  $m$  items. Each bidder  $i$  has a private valuation function  $v_i: 2^M \rightarrow \mathbb{R}_{\geq 0}$ , defining a non-negative value of each subset of items. Let  $V_i$  denote the set of all valuation functions. This set might be restricted to particular functions in certain settings. An interesting class of valuation functions are *unit-demand* functions, for which there are  $v_{i,j}$  such that  $v_i(S) = \max_{j \in S} v_{i,j}$ . That is, each bidder's valuation is only the maximum value of a single item in the set.

The set of feasible allocations is given as  $X = \{(S_1, \dots, S_n) \mid S_i \cap S_{i'} = \emptyset \text{ for } i \neq i'\}$ .

A direct mechanism is a pair  $\mathcal{M} = (f, p)$ , consisting of an allocation function  $f: V \rightarrow X$  and a payment rule  $p: V \rightarrow \mathbb{R}^n$ , where  $V = V_1 \times \dots \times V_n$ .

**Example 13.1.** We could have  $N = \{1, 2\}$ ,  $M = \{A, B\}$ . The following valuations are unit-demand:  $v_1(\{A\}) = 5$ ,  $v_1(\{B\}) = 9$ ,  $v_1(\{A, B\}) = 9$ ,  $v_2(\{A\}) = 0$ ,  $v_2(\{B\}) = 5$ ,  $v_2(\{A, B\}) = 5$ ,  $v_1(\emptyset) = v_2(\emptyset) = 0$ .

As there is no additional value from giving a bidder more than one item, the problem of maximizing social welfare corresponds to finding the max-weight matching in the following graph (indicated by bold edges).



## 2 Model

Combinatorial auctions are just one example of a problem that fits into our general model. We may have an arbitrary set of feasible outcomes  $X$ . For each bidder  $i$ , there is a set  $V_i$  of possible valuation functions  $v_i: X \rightarrow \mathbb{R}$ . (Syntactically this is a little different and more general than the valuations in the combinatorial auctions above.) We denote  $V = V_1 \times \dots \times V_n$ .

A direct mechanism is a pair  $\mathcal{M} = (f, p)$ , consisting of an allocation function  $f: V \rightarrow X$  and a payment rule  $p: V \rightarrow \mathbb{R}^n$ . Bidder  $i$ 's utility under bids  $b \in V$  is given by  $u_i(b, v_i) = v_i(f(b)) - p_i(b)$ .

## 3 VCG Mechanism with Clarke Pivot Rule

As a matter of fact, sometimes people refer to VCG mechanisms as a class of mechanisms following a particular template. However, whenever one says “the” VCG mechanism, this will refer to the following VCG mechanism with Clarke pivot rule.

**Definition 13.2.** Let  $f: V \rightarrow X$  be a function that maximizes declared welfare, that is  $f(b) \in \arg \max_{x \in X} \sum_i b_i(x)$  for all  $b \in V$ . Then the VCG mechanism with Clarke pivot rule is defined as  $\mathcal{M} = (f, p)$ , where

$$p_i(b) = \max_{x \in X} \sum_{j \neq i} b_j(x) - \sum_{j \neq i} b_j(f(b)) .$$

The idea behind this payment rule is as follows. The first sum represents the declared welfare of all bidders except for  $i$  that would be achieved if we were not restricted in any way. The second term is exactly the amount that is achieved by  $f(b)$ , which means that we are optimizing over all bidders including  $i$ . The difference therefore is the loss of declared welfare due to the presence of bidder  $i$ . This is called bidder  $i$ 's externality.

**Theorem 13.3** (Vickrey-Clarke-Groves). *The VCG mechanism with Clarke pivot rule is dominant-strategy incentive compatible.*

*Proof.* Observe that for all  $b_i, b_{-i}$ ,

$$u_i((b_i, b_{-i}), v_i) = v_i(f(b_i, b_{-i})) - p_i(b_i, b_{-i}) = v_i(f(b_i, b_{-i})) - \max_{x \in X} \sum_{j \neq i} b_j(x) + \sum_{j \neq i} b_j(f(b_i, b_{-i})) .$$

On input  $(v_i, b_{-i})$ , the function  $f$  returns a solution  $x^*$ , which maximizes  $v_i(x^*) + \sum_{j \neq i} b_j(x^*)$ . That is, for any  $x \in X$ , we have  $v_i(x^*) + \sum_{j \neq i} b_j(x^*) \geq v_i(x) + \sum_{j \neq i} b_j(x)$ . In particular, this holds for  $x = f(b_i, b_{-i})$  for all possible  $b_i$ .

Consequently,

$$v_i(f(v_i, b_{-i})) + \sum_{j \neq i} b_j(f(v_i, b_{-i})) \geq v_i(f(b_i, b_{-i})) + \sum_{j \neq i} b_j(f(b_i, b_{-i}))$$

and therefore

$$u_i((v_i, b_{-i}), v_i) \geq u_i((b_i, b_{-i}), v_i) . \quad \square$$

## 4 Examples

### 4.1 Single-Item Auctions Revisited

As a first example for VCG, let us consider single-item auctions again. Remember that each agent's valuation function  $v_i$  given by

$$v_i(x) = \begin{cases} t_i & \text{if agent } i \text{ receives the item in } x \\ 0 & \text{otherwise} . \end{cases}$$

Given the vector  $b$ , the function  $f$  selects the agent with the highest bid. Let this agent be denoted by  $i^*$ . For  $i^*$ , we now have

$$p_{i^*}(b) = \max_{x \in X} \sum_{j \neq i^*} b_j(x) - \sum_{j \neq i^*} b_j(f(b)) .$$

We have  $\max_{x \in X} \sum_{j \neq i^*} b_j(x)$  is exactly the second-highest bid. Furthermore, for  $j \neq i^*$ , we have  $b_j(f(b)) = 0$  because agent  $j$  does not get the item.

For all agents  $i \neq i^*$

$$p_i(b) = \max_{x \in X} \sum_{j \neq i} b_j(x) - \sum_{j \neq i} b_j(f(b)) = b_{i^*} - b_{i^*} = 0 .$$

That is, agent  $i^*$  pays the second highest bid, the other agents pay nothing. This is exactly the second-price auction.

### 4.2 Sponsored Search Auctions

In a sponsored search auction, we sell  $k < n$  ad slots on a search results page. The higher the slot is displayed on the page, the more likely it will be clicked. For slots  $1, \dots, k$ , we assume click through rates of  $\alpha_1 \geq \alpha_2 \geq \dots \geq \alpha_k$ . Agent  $i$ 's valuation is expressed in terms of a single number  $v_i$  such that  $v_i(x) = v_i \alpha_j$  if agent  $i$  gets slot  $j$  in  $x$ .

If  $v_1 \geq v_2 \geq \dots \geq v_n$ , then the social-welfare optimizing allocation gives slot  $j$  to bidder  $j$ . This results in social welfare  $\sum_{j=1}^k v_j \alpha_j$ . The optimal social welfare without agent  $i$  is  $\sum_{j=1}^{i-1} v_j \alpha_j + \sum_{j=i+1}^{k+1} v_j \alpha_{j-1}$ . Consequently, given truthful reports, if we set  $\alpha_{k+1} = 0$ , agent  $i$ 's VCG payment is

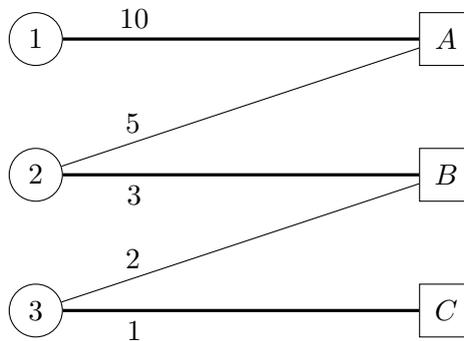
$$p_i(v) = \sum_{j=1}^{i-1} v_j \alpha_j + \sum_{j=i+1}^{k+1} v_j \alpha_{j-1} - \left( \sum_{j=1}^{i-1} v_j \alpha_j + \sum_{j=i+1}^k v_j \alpha_j \right) = \sum_{j=i+1}^{k+1} v_j (\alpha_{j-1} - \alpha_j) .$$

Interestingly, for mysterious reasons in practice this scheme is not applied. Instead a rule called *generalized second price* is used: Agent  $i$  has to pay  $v_{i+1} \alpha_{i+1}$ . This is generally not incentive compatible.

### 4.3 Unit-Demand Combinatorial Auction

Let us come back to our initial example of unit-demand combinatorial auctions. That is, there are  $m$  items  $M$  and each bidder's valuation is of the form  $v_i(S) = \max_{j \in S} v_{i,j}$ . Maximizing the declared welfare corresponds to finding the maximum-weight matching in a complete bipartite graphs whose vertices are  $N \cup M$ . The edge between  $i \in N$  and  $j \in M$  has weight  $v_{i,j}$ .

Let us consider an example with three bidders 1, 2, 3 and three items  $A, B, C$ . We only draw edges of positive value.



The optimal matching is given by the thick edges. The payments are computed by removing the respective bidder vertex and re-optimizing. Consequently, the payment of a bidder is given by the most valuable augmenting path that arises by removing him. In the above example, bidder 1 has to pay  $5 - 3 + 2 - 1 = 3$ .

## 5 Further Properties

Besides incentive compatibility, the mechanism also enjoys the following nice properties:

- **Individual Rationality.** If  $v_i(x) \geq 0$  for all  $x$ , then  $u_i((v_i, b_{-i}), v_i) \geq 0$  for all  $b_{-i}$ . The reason is that

$$\begin{aligned} u_i((v_i, b_{-i}), v_i) &= v_i(f(v_i, b_{-i})) + \sum_{j \neq i} b_j(f(v_i, b_{-i})) - \max_{x \in X} \sum_{j \neq i} b_j(x) \\ &= \left( \max_{x \in X} \left( v_i(x) + \sum_{j \neq i} b_j(x) \right) \right) - \left( \max_{x \in X} \sum_{j \neq i} b_j(x) \right) \geq 0 . \end{aligned}$$

The term is non-negative because  $v_i(x) + \sum_{j \neq i} b_j(x) \geq \sum_{j \neq i} b_j(x)$  for all  $x$ . Therefore this also holds for the maximum.

- **No Positive Transfer.** For all  $b$ , we have

$$p_i(b) = \left( \max_{x \in X} \sum_{j \neq i} b_j(x) \right) - \left( \sum_{j \neq i} b_j(f(b)) \right) \geq 0 ,$$

because  $\sum_{j \neq i} b_j(f(b)) \leq \max_{x \in X} \sum_{j \neq i} b_j(x)$ : The left-hand side is just one possible value that this expression can take whereas it is maximized on the right-hand side.

## 6 Limitations

We have seen that VCG mechanisms work well in many environments. However, it does not yet solve all questions regarding mechanism design with money. There are several limitations: First of all, to build a VCG mechanism, we have to solve the welfare-maximization problem optimally. In many cases, this problem is actually intractable. Below we will see that only approximating social welfare is not enough. VCG also does not optimize the payments in any sense. For example, it does not even try to maximize the revenue obtained by the payments. Also, agents only have a limited budget, but we do not ensure that they only spend a certain amount. Finally, it might be a problem that agents collude. Although each single agents cannot benefit from false reports themselves, other agents can.

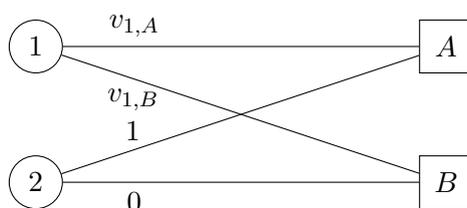
Probably the biggest limitation from an algorithmic aspect is the fact that VCG requires a welfare-maximizing solution. It will be instructive to see that this is indeed necessary because there are approximation algorithms that cannot be turned into an incentive compatible mechanism.

To show this, we again consider unit-demand combinatorial auctions. A fast way to find a reasonable matching is the greedy algorithm: Consider all pairs  $(i, j)$  by non-increasing  $b_{i,j}$ . Assign item  $j$  to bidder  $i$  unless item  $j$  has already been allocated or bidder  $i$  has already received an item.

This is a very natural algorithm. It does not always compute the allocation that maximizes declared welfare but it is 2-approximation. That is, we have  $\sum_i b_i(f(b)) \geq \frac{1}{2} \max_{x \in S} \sum_i b_i(x)$  for all  $b$ . However, VCG payments won't turn it into a truthful mechanism. We will show this even in a more general way: There is no payment rule that turns this into a truthful mechanism.

**Theorem 13.4.** *Let  $f$  be the greedy algorithm for unit-demand combinatorial auctions. There is no payment rule  $p$  such that  $\mathcal{M} = (f, p)$  is a truthful mechanism.*

*Proof.* We consider this kind of instance to show that no payment scheme can render the mechanism incentive compatible.



There are two items  $A$  and  $B$ . Bidder 1 has values  $v_{1,A}$  and  $v_{1,B}$ ; bidder 2 has values 1 and 0. We keep bidder 2's valuation and report fixed at all times.

For bidder 1, we compare three possible valuation functions  $v_1^{(1)}$ ,  $v_1^{(2)}$ ,  $v_1^{(3)}$ , defined by

$$\begin{array}{ll} v_{1,A}^{(1)} = 1 - \epsilon & v_{1,B}^{(1)} = \epsilon \\ v_{1,A}^{(2)} = 1 + \epsilon & v_{1,B}^{(2)} = 1 + 2\epsilon \\ v_{1,A}^{(3)} = 1 + 2\epsilon & v_{1,B}^{(3)} = 1 + \epsilon \end{array}$$

where  $\epsilon < \frac{1}{3}$ .

Suppose, we had a truthful mechanism  $\mathcal{M} = (f, p)$ . Then, by truthfulness,

$$u_1((v_1^{(2)}, v_2), v_1^{(2)}) \geq u_1((v_1^{(1)}, v_2), v_1^{(2)})$$

because bidder 1 should not benefit from misreporting  $v_1^{(1)}$  when her true valuation is  $v_1^{(2)}$ . In both cases, bidder 1 is getting item  $B$ , so

$$1 + 2\epsilon - p_1(v_1^{(2)}, v_2) \geq 1 + 2\epsilon - p_1(v_1^{(1)}, v_2) ,$$

which implies  $p_1(v_1^{(2)}, v_2) \leq p_1(v_1^{(1)}, v_2)$ .

Analogously, we get

$$u_1((v_1^{(3)}, v_2), v_1^{(3)}) \geq u_1((v_1^{(2)}, v_2), v_1^{(3)}) .$$

Note that in  $(v_1^{(3)}, v_2)$ , bidder 1 will get item  $A$ , in  $(v_1^{(2)}, v_2)$  she will get item  $B$ . So

$$1 + 2\epsilon - p_1(v_1^{(3)}, v_2) \geq 1 + \epsilon - p_1(v_1^{(2)}, v_2)$$

and therefore  $p_1(v_1^{(3)}, v_2) \leq p_1(v_1^{(2)}, v_2) + \epsilon$ .

Now, we come to the case in which the true valuation in  $v_1^{(1)}$ . For truthfulness, we need

$$u_1((v_1^{(1)}, v_2), v_1^{(1)}) \geq u_1((v_1^{(3)}, v_2), v_1^{(1)}) .$$

When truthfully bidding  $v_1^{(1)}$ , bidder 1 will receive item  $B$ . When bidding  $v_1^{(3)}$ , she will receive item  $A$ . This would mean that

$$\epsilon - p_1(v_1^{(1)}, v_2) \geq 1 - \epsilon - p_1(v_1^{(3)}, v_2) ,$$

which implies  $p_1(v_1^{(3)}, v_2) \geq p_1(v_1^{(1)}, v_2) + 1 - 2\epsilon$ .

This, however, cannot be possible if also  $p_1(v_1^{(3)}, v_2) \leq p_1(v_1^{(2)}, v_2) + \epsilon \leq p_1(v_1^{(1)}, v_2) + \epsilon$  and  $\epsilon < \frac{1}{3}$ .  $\square$

## References and Further Reading

- Chapter 9.3 in the AGT book
- Tim Roughgarden's lecture notes <http://timroughgarden.org/f13/1/17.pdf> and lecture video <https://youtu.be/TL13FVXPVIY>
- Eva Tardos's lecture notes <http://www.cs.cornell.edu/courses/cs6840/2012sp/lec18.pdf>
- Original papers:
  - W. Vickrey. Counterspeculation, Auctions, and Competitive Sealed Tenders. *Journal of Finance*. 16 (1): 8–37, 1961.
  - E. Clarke. Multipart Pricing of Public Goods. *Public Choice*. 11 (1): 17–33, 1971
  - T. Groves. Incentives in Teams. *Econometrica*. 41 (4): 617–631, 1973.