Lecture 7

Multi-Parameter Mechanism Design

Lectures 2–6 considered only single-parameter mechanism design problems, where the only private parameter of an agent is her valuation per unit of stuff. Mechanism design is much more difficult for *multi-parameter* problems, where each agent has multiple private parameters. The Vickrey-Clarke-Groves (VCG) mechanisms provide a sweeping positive result: DSIC welfare maximization is possible in principle in every multi-parameter environment.

Section 7.1 formally defines general mechanism design environments. Section 7.2 derives the VCG mechanisms and proves that they are DSIC. Section 7.3 discusses the challenges of implementing VCG mechanisms in practice.

7.1 General Mechanism Design Environments

A general multi-parameter mechanism design environment comprises the following ingredients:

- n strategic participants, or "agents";
- a finite set Ω of outcomes;
- each agent i has a private nonnegative valuation $v_i(\omega)$ for each outcome $\omega \in \Omega$.

The outcome set Ω is abstract and could be very large. The *social welfare* of an outcome $\omega \in \Omega$ is defined as $\sum_{i=1}^{n} v_i(\omega)$.

Example 7.1 (Single-Item Auction Revisited) In a single-item auction, Ω has only $n+1$ elements, corresponding to the winner of the item (if any). In the standard single-parameter model of a single-item auction, we assume that the valuation of a bidder is 0 in all of the n outcomes in which she doesn't win, leaving only one unknown parameter per bidder. In the more general multi-parameter framework, a bidder can have a different valuation for each possible winner of the auction. For example, in a bidding war over a hot startup, if a bidder loses, she might prefer that the startup be bought by a company in a different market, rather than by a direct competitor.

Example 7.2 (Combinatorial Auctions) In a combinatorial auction, there are multiple indivisible items for sale, and bidders can have complex preferences between different subsets of items (called *bundles*). With n bidders and a set M of m items, the outcomes of Ω correspond to n-vectors $(S_1, \ldots, S_n)$, with $S_i \subseteq M$ denoting the bundle allocated to bidder i, and with no item allocated twice. There are $(n+1)^m$ different outcomes. Each bidder i has a private valuation $v_i(S)$ for each bundle $S \subseteq M$ of items she might get. Thus, each bidder has 2^m private parameters.

Combinatorial auctions are important in practice. For example, dozens of government spectrum auctions around the world have raised hundreds of billions of dollars of revenue. In such auctions, typical bidders are telecommunication companies like Verizon or AT&T, and each item is a license awarding the right to broadcast on a certain frequency in a given geographic area. Combinatorial auctions have also been used for other applications such as allocating takeoff and landing slots at airports.

7.2 The VCG Mechanism

Our next result is a cornerstone of mechanism design theory, and one of the most sweeping positive results in the field: *every* multi-parameter environment admits a DSIC welfare-maximizing mechanism.

Theorem 7.3 (Multi-Parameter Welfare Maximization) *In every general mechanism design environment, there is a DSIC welfare-maximizing mechanism.*

Recall the three properties of ideal mechanisms that we singled out in Theorem 2.4, in the context of second-price auctions. Theorem 7.3 asserts the first two properties (DSIC and welfare maximization) but not the third (computational efficiency). We already know from Section 4.1.4 that, even in single-parameter environments, we can't always have the second and third properties (unless $\mathcal{P} = \mathcal{NP}$). We'll see that the mechanism identified in Theorem 7.3 is highly non-ideal in many important applications.

We discuss the main ideas behind Theorem 7.3 before proving it formally. Designing a (direct-revelation) DSIC mechanism is tricky because the allocation and payment rules need to be coupled carefully.[1] We apply the same two-step approach that served us so well in single-parameter environments (Section 2.6.4).

The first step is to assume, without justification, that agents truthfully report their private information, and then figure out which outcome to pick. Since Theorem 7.3 demands welfare maximization, the only solution is to pick a welfare-maximizing outcome, using bids as proxies for the unknown valuations. That is, given reports $\mathbf{b}_1, \ldots, \mathbf{b}_n$, where each report $\mathbf{b}_i$ is now a vector indexed by Ω, we define the allocation rule $\mathbf{x}$ by

$$\mathbf{x}(\mathbf{b}) = \operatorname{argmax}_{\omega \in \Omega} \sum_{i=1}^{n} b_i(\omega). \tag{7.1}$$

The second step is to define a payment rule that, when coupled with this allocation rule, yields a DSIC mechanism. Last time we faced this problem (Section 3.3), for single-parameter environments, we formulated and proved Myerson's lemma (Theorem 3.7), which is a general solution to this second step for all such environments. Myerson's lemma does not hold beyond single-parameter environments—with each agent submitting a multidimensional report, it's not even clear how to define "monotonicity" of an allocation rule (cf., Definition 3.6).[2] Similarly, the "critical bid" characterization of DSIC

[1] The statement and proof of the revelation principle (Theorem 4.3) extend immediately to general mechanism design environments, so we can restrict attention to direct-revelation mechanisms without loss of generality.

[2] There is an analogous characterization of the implementable multi-parameter allocation rules in terms of "cycle monotonicity"; see the Notes. This is an elegant result, analogous to the fact that a network with real-valued lengths on its edges admits well-defined shortest paths if and only if it possesses no negative cycle. Cycle monotonicity is far more unwieldy than single-parameter monotonicity, however. Because it is so brutal to verify, cycle monotonicity is rarely used to argue implementability or to derive DSIC payment rules in concrete settings.

payments for 0-1 single-parameter problems (Section 4.1.3) does not have an obvious analog in multi-parameter problems.

The key idea is to generalize an alternative characterization of an agent i's payment in a DSIC welfare-maximizing mechanism, as the "externality" caused by i— the welfare loss inflicted on the other $n-1$ agents by i's presence (cf., Exercise 4.2). For example, in a single-item auction, the winning bidder inflicts a welfare loss on the others equal to the second-highest bid (assuming truthful bids), and this is precisely the payment rule of a second-price auction. "Charging an agent her externality" remains well defined in general mechanism design environments, and corresponds to the payment rule

$$p_i(\mathbf{b}) = \underbrace{\left(\max_{\omega \in \Omega} \sum_{j \neq i} b_j(\omega)\right)}_{\text{without } i} - \underbrace{\sum_{j \neq i} b_j(\omega^*)}_{\text{with } i}, \tag{7.2}$$

where $\omega^* = \mathbf{x}(\mathbf{b})$ is the outcome chosen in (7.1). Intuitively, these payments force the agents to "internalize" their externalities, thereby aligning their incentives with those of a welfare-maximizing decision maker. The payment $p_i(\mathbf{b})$ is always at least 0 (Exercise 7.1).

Definition 7.4 (VCG Mechanism) A mechanism $(\mathbf{x}, \mathbf{p})$ with allocation and payment rules as in (7.1) and (7.2), respectively, is a *Vickrey-Clarke-Groves* or *VCG* mechanism.

For an alternative interpretation of the payments in a VCG mechanism, rewrite the expression in (7.2) as

$$p_i(\mathbf{b}) = \underbrace{b_i(\omega^*)}_{\text{bid}} - \underbrace{\left[\sum_{j=1}^{n} b_j(\omega^*) - \max_{\omega \in \Omega} \sum_{j \neq i} b_j(\omega)\right]}_{\text{rebate}}. \tag{7.3}$$

We can therefore think of agent i's payment as her bid minus a "rebate," equal to the increase in welfare attributable to i's presence. For example, in a second-price auction, the highest bidder pays her bid b_1 minus a rebate of $b_1 - b_2$ (where b_2 is the second-highest bid), the increase in welfare that the bidder brings to the table. With nonnegative reports, the rebate in (7.3) is nonnegative (Exercise 7.1).

This implies that $p_i(\mathbf{b}) \leq b_i(\omega^*)$ and hence truthful reporting always guarantees nonnegative utility.

Proof of Theorem 7.3: Fix an arbitrary general mechanism design environment and let $(\mathbf{x}, \mathbf{p})$ denote the corresponding VCG mechanism. By definition, the mechanism maximizes the social welfare whenever all reports are truthful. To verify the DSIC condition (Definition 2.3), we need to show that for every agent i and every set $\mathbf{b}_{-i}$ of reports by the other agents, agent i maximizes her quasilinear utility $v_i(\mathbf{x}(\mathbf{b})) - p_i(\mathbf{b})$ by setting $\mathbf{b}_i = \mathbf{v}_i$.

Fix i and $\mathbf{b}_{-i}$. When the chosen outcome $\mathbf{x}(\mathbf{b})$ is ω^*, we can use (7.2) to write i's utility as

$$v_i(\omega^*) - p_i(\mathbf{b}) = \underbrace{\left[v_i(\omega^*) + \sum_{j \neq i} b_j(\omega^*) \right]}_{\text{(A)}} - \underbrace{\left[\max_{\omega \in \Omega} \sum_{j \neq i} b_j(\omega) \right]}_{\text{(B)}}.$$

The term (B) is a constant, independent of i's report $\mathbf{b}_i$. Thus, the problem of maximizing agent i's utility reduces to the problem of maximizing the first term (A). As a thought experiment, suppose agent i has the power to choose the outcome ω^* directly, rather than merely influencing the chosen outcome indirectly via her choice of bid $\mathbf{b}_i$. Agent i would, of course, use this extra power to choose an outcome that maximizes the term (A). If agent i sets $\mathbf{b}_i = \mathbf{v}_i$, then the function (7.1) that the mechanism maximizes becomes identical to the term (A) that the agent wants maximized. Thus, truthful reporting coaxes the mechanism to choose an outcome that maximizes agent i's utility; no other report can be better. ■

7.3 Practical Considerations

Theorem 7.3 shows that, in general multi-parameter environments, DSIC welfare maximization is always possible in principle. However, there are several major obstacles to implementing the VCG mechanism in most multi-parameter environments.[3]

[3]The VCG mechanism can still serve as a useful benchmark for other, more practical solutions (cf., Lecture 6).

The first challenge of implementing the VCG mechanism is *preference elicitation*, meaning getting the reports $\mathbf{b}_1, \ldots, \mathbf{b}_n$ from the agents. For example, in a combinatorial auction with m items (Example 7.2), each bidder has 2^m private parameters, which is roughly a thousand when $m = 10$ and a million when $m = 20$. No bidder in her right mind would want to figure out or write down so many numbers, and no seller would want to read them. This critique applies to every direct-revelation mechanism, not just to the VCG mechanism, for an environment with a large outcome space.

The second challenge is familiar from algorithmic mechanism design (Lecture 4). Even when the first challenge is not an issue and preference elicitation is easy, as in single-parameter environments, welfare maximization can be a computationally intractable problem. This is already the case in (single-parameter) knapsack auctions (Section 4.1.4), and in more complex settings even approximate welfare maximization can be computationally intractable.

The third challenge is that, even in applications where the first two challenges are not relevant, VCG mechanisms can have bad revenue and incentive properties (despite being DSIC). For instance, consider a combinatorial auction with two bidders and two items, A and B. The first bidder only wants both items, so $v_1(AB) = 1$ and is 0 otherwise. The second bidder only wants item A, so $v_2(AB) = v_2(A) = 1$ and is 0 otherwise. The revenue of the VCG mechanism is 1 in this example (Exercise 7.2). Now suppose we add a third bidder who only wants item B, so $v_3(AB) = v_3(B) = 1$. The maximum welfare jumps to 2, but the VCG revenue drops to 0 (Exercise 7.2)! The fact that the VCG mechanism has zero revenue in seemingly competitive environments is a dealbreaker in practice. The revenue non-monotonicity in this example also leads to numerous incentive problems, including vulnerability to collusion and false-name bids (Exercises 7.3 and 7.4).

The next lecture discusses how practitioners cope with these challenges in real-world combinatorial auctions.

The Upshot

☆ In a general mechanism design environment, each agent has a private valuation for each possible outcome. Combinatorial auctions are an important example in both theory and practice.

☆ In the VCG mechanism for an environment, the allocation rule selects an outcome that maximizes the social welfare with respect to agents' reports. The payment rule charges each agent her externality—the welfare loss inflicted on the other agents by her presence.

☆ Every VCG mechanism is DSIC.

☆ There are several obstacles to implementing VCG mechanisms in practice, including the difficulty of eliciting a large number of private parameters, the computational intractability of computing a welfare-maximizing outcome, and their bad revenue and incentive properties.

Notes

Our definition of VCG mechanisms follows Clarke (1971), generalizing the second-price single-item auction of Vickrey (1961). Groves (1973) gives a still more general class of mechanisms, where a bid-independent "pivot term" $h_i(\mathbf{b}_{-i})$ is added to each agent's payment, as in Problem 7.1. The equivalence of multi-parameter implementability and "cycle monotonicity" is due to Rochet (1987); see Vohra (2011) for a lucid exposition and some applications. Rothkopf et al. (1990) and Ausubel and Milgrom (2006) detail the many challenges of implementing VCG mechanisms in practice. Problem 7.1 is due to Holmstrom (1977). Problem 7.3 is from Dobzinski et al. (2010); see Blumrosen and Nisan (2007) for a survey of further results of this type.

Exercises

Exercise 7.1 Prove that the payment $p_i(\mathbf{b})$ charged to an agent i in the VCG mechanism is at least 0 and at most $b_i(\omega^*)$, where ω^* is the outcome chosen by the mechanism.

Exercise 7.2 Consider a combinatorial auction (Example 7.2) with two items, A and B, and three bidders. The first bidder has valuation 1 for receiving both items (i.e., $v_1(AB) = 1$) and 0 otherwise. The second bidder has valuation 1 for item A (i.e., $v_2(AB) = v_2(A) = 1$) and 0 otherwise. The third bidder has valuation 1 for B and 0 otherwise.

(a) Compute the outcome of the VCG mechanism when only the first two bidders are present and when all three bidders are present. What can you conclude?

(b) Can adding an extra bidder ever decrease the revenue of a second-price single-item auction?

Exercise 7.3 Exhibit a combinatorial auction and bidder valuations such that the VCG mechanism has the following property: there are two bidders who receive no items when all bidders bid truthfully, but can both achieve positive utility by submitting suitable false bids (assuming others bid truthfully). Why doesn't this example contradict Theorem 7.3?

Exercise 7.4 Consider a combinatorial auction in which bidders can submit multiple bids under different names, unbeknownst to the mechanism. The allocation and payment of a bidder is the union and sum of the allocations and payments, respectively, assigned to all of her pseudonyms.

(a) Exhibit a combinatorial auction and bidder valuations such that, in the VCG mechanism, there is a bidder who can earn higher utility by submitting multiple bids than by bidding truthfully as a single agent (assuming others bid truthfully).

(b) Can this ever happen in a second-price single-item auction?

Exercise 7.5 *(H)* A bidder i in a combinatorial auction has a *unit-demand* valuation if there exist parameters $v_{i1}, \ldots, v_{im}$, one per item, such that $v_i(S) = \max_{j \in S} v_{ij}$ for every bundle S of items (and $v_i(\emptyset) = 0$). A bidder with a unit-demand valuation only wants one item—for instance, a hotel room for a given night—and only retains her favorite item from a bundle.

Give an implementation of the VCG mechanism in combinatorial auctions with unit-demand bidder valuations that runs in time polynomial in the number of bidders and the number of items.

Problems

Problem 7.1 Consider a general mechanism design environment, with outcome set Ω and n agents. In this problem, we use the term DSIC to refer to the first condition of Definition 2.3 (truthful reporting is a dominant strategy) and do not consider the individual rationality condition (truthful bidders receive nonnegative utility).

(a) Suppose we modify the payments (7.2) of the VCG mechanism by adding a *pivot term* $h_i(\mathbf{b}_{-i})$ to each agent i's payment, where $h_i(\cdot)$ is an arbitrary function of the other agents' reports. These pivot terms can be positive or negative, and can result in payments from the mechanism to the agents. Prove that for every choice of pivot terms, the resulting mechanism is DSIC.

(b) *(H)* Suppose agents' valuations are restricted to lie in the set $\mathcal{V} \subseteq \mathbb{R}^{\Omega}$. We say that the pivot terms $\{h_i(\cdot)\}_{i=1}^{n}$ *budget-balance the VCG mechanism* if, for all possible reports $b_1(\cdot), \ldots, b_n(\cdot) \in \mathcal{V}$, the corresponding VCG payments (including the h_i's) sum to 0.

Prove that there exist pivot terms that budget-balance the VCG mechanism if and only if the maximum social welfare can be represented as the sum of bid-independent functions—if and only if we can write

$$\max_{\omega \in \Omega} \sum_{i=1}^{n} b_i(\omega) = \sum_{i=1}^{n} g_i(\mathbf{b}_{-i}) \tag{7.4}$$

for every $b_1(\cdot), \ldots, b_n(\cdot) \in \mathcal{V}$, where each g_i is a function that does not depend on b_i.

(c) Either directly or using part (b), prove that there are no pivot terms that budget-balance a second-price single-item auction.

Conclude that there is no DSIC single-item auction that maximizes social welfare and is budget-balanced.

Problem 7.2 Consider a general mechanism design environment, with outcome set Ω and n agents. Suppose the function $f : \Omega \to \mathbb{R}$ has the form

$$f(\omega) = c(\omega) + \sum_{i=1}^{n} w_i v_i(\omega),$$

where c is a publicly known function of the outcome, and where each w_i is a nonnegative, public, agent-specific weight. Such a function is called an *affine maximizer*.

Show that for every affine maximizer f and every subset $\Omega' \subseteq \Omega$ of the outcomes, there is a DSIC mechanism that maximizes f over Ω'.

Problem 7.3 Consider a combinatorial auction (Example 7.2) with a set M of m items, where the valuation function $v_i : 2^M \to \mathbb{R}^+$ of each bidder i satisfies: (i) $v_i(\emptyset) = 0$; (ii) $v_i(S) \leq v_i(T)$ whenever $S \subseteq T \subseteq M$; and (iii) $v_i(S \cup T) \leq v_i(S) + v_i(T)$ for all bundles $S, T \subseteq M$. Such functions are called *subadditive*.

(a) Call a profile $\mathbf{v}$ of subadditive valuations *lopsided* if there is a social welfare-maximizing allocation in which at least 50% of the social welfare is contributed by bidders who are each allocated at least $\sqrt{m}$ items. Prove that if $\mathbf{v}$ is lopsided, then there is an allocation that gives all of the items to a single bidder and has social welfare at least $\frac{1}{2\sqrt{m}}$ times the maximum possible.

(b) *(H)* Prove that if $\mathbf{v}$ is not lopsided, then there is an allocation that gives at most one item to each bidder and has social welfare at least $\frac{1}{2\sqrt{m}}$ times the maximum possible.

(c) *(H)* Give a mechanism with the following properties: (1) for some collection $\mathcal{S}$ of bundles, with $|\mathcal{S}|$ polynomial in m, each bidder i submits a bid $b_i(S)$ only on the bundles $S \in \mathcal{S}$; (2) for every bidder i and bids by the others, it is a dominant strategy to set $b_i(S) = v_i(S)$ for all $S \in \mathcal{S}$; (3) assuming truthful bids, the outcome of the mechanism has social welfare at least $\frac{1}{2\sqrt{m}}$ times the maximum possible; (4) the running time of the mechanism is polynomial in m and the number n of bidders.