Mechanism Design Basics: Auction Theory

September 3, 2020

Denizalp Goktas

1. Need to add KC, CK, clinching, DCA and unit demand business somewhere
2. would be nice to add Knapsack somewhere but not important

1 Single-Item Auction

We start our discussion of mechanism design with a simple allocation problem, that of the single-item auction. We first describe the single-item auction model, analy

1.1 Model

The model consists of:

1. One good/item that is indivisible, i.e., the item can only be allocated to one bidder.
2. A set of n bidders/buyers
3. Each bidder $i \in [n]$ has:
   - a private valuation v_i which corresponds to the maximum amount that bidder i is willing to pay for the auctioned good
   - a utility function $u_i(x_i; p_i) = x_i(v_i - p_i)$ where x_i is a binary variable denoting whether if the bidder i was allocated the good or not.

1.2 Auction Mechanism

A **direct single-item auction mechanism** $\mathcal{A} : \mathbb{R}^n \to \{0,1\}^n \times \mathbb{R}$ is a function that takes as input the **reported valuations/bids** of the bidders and outputs a tuple consisting of an allocation vector $\boldsymbol{x} \in \{0,1\}^n$ denoting the allocation of the auctioned good to i^{th} bidders $i \in [n]$ and a price $\boldsymbol{p} \in \mathbb{R}^n$. It is important to note that the bidders can report bids strategically, i.e., they can lie about their valuations to potentially improve the outcome for themselves.

An **indirect single-item auction mechanism** $\mathcal{A} : \boldsymbol{A}^n \times \mathbb{R}$ is a function that takes as input the actions of the bidders which come from the action space $\boldsymbol{A}$ and outputs a tuple consisting of an allocation vector $\boldsymbol{x} \in \{0,1\}^n$ denoting the allocation of the auctioned good to bidders $i \in [n]$ and a payment rule $\boldsymbol{p} \in \mathbb{R}^n$.

Often, we will denote the allocation rule outputted by an auction mechanism $\mathcal{A}$ as a function of the bids, $\boldsymbol{b}$, by $\boldsymbol{x}(\boldsymbol{b})$, and the payment rule as a function $p(\boldsymbol{b})$.

Additionally, note that the phrase “direct mechanism" refers to mechanisms that elicit information from the bidders that is directly tied to their private information (in this case the bids), while an indirect mechanism elicits information that is indirectly tied to the private information of the bidders. We will see some examples which will clarify the distinction better.

1.3 Properties of Outcomes

Different single-item auction mechanisms can have different properties. We define some of these properties.

Incentive Compatibility (IC): A single-item auction mechanism $\mathcal{A}$ is said to be incentive compatible iff the bidder cannot improve his utility by lying about his private valuation.

Individually Rational (IR): A single item auction mechanism $\mathcal{A}$ is said to be individually rational iff for all bidders reporting their private valuation v_i to the mechanism $\mathcal{A}$ does not result in an outcome that gives them negative utility.

Welfare Maximizing (WM): A single-item auction mechanism $\mathcal{A}$ is said to be welfare maximizing iff the allocation outputted by $\mathcal{A}$ maximizes the surplus of the auctioneer and the bidders $\sum_{i\in[n]} x_i(v_i-p)+p = \sum_{i\in[n]} x_i v_i$.

1.4 Mechanism Examples

In this section, we present a few mechanism for single-item auctions. First, we discuss direct mechanisms and then indirect mechanisms.

1.5 Direct Mechanisms

A **first price auction** is an auction mechanism that assigns the auctioned good to the highest bidder and asks the winner to pay its bid.

A **second price auction/Vickrey Auction** is an auction that assigns the auctioned good to the highest bidder and asks the winner to pay the second highest bidder's bid.

Theorem 1.1. *The Vickrey auction is 1) incentive compatible, 2) individually rational and 3) welfare maximizing. Furthermore, it can be implemented in polynomial (linear) time.*

Theorem 1.2. *If bidders' private values are uniform i.i.d., i.e, $\forall i \in [n], v_i \sim Unif[0,1]$, then the expected revenue (i.e., expected price of the auctioned good) of the first-price auction is equal to that of the second-price auction assuming that bidders behave according to their respective equilibrium strategies*

1.6 Indirect Mechanisms

One example of an indirect mechanism is the English Auction, which we introduce now.

The English auction consists of a number of rounds. On round $k = 1, 2, \ldots,$ the auctioneer offers the good at price $p = k\epsilon$ asking all bidders if they are interested in the good at that price. The auction continues so long as more than one bidder is interested. The auction terminates, say at round t, when either exactly one or no bidders remain interested. If there is one interested bidder, then she wins, paying $t\epsilon$; if there are no interested bidders then a winner is selected at random from the set of interested bidders during round $t-1$. This winner pays $(t-1)\epsilon$

In this auction, actions consist of t binary answers to queries "Would you like the good at price p?" It may be easier to answer these questions to identify an exact valuation, which is why this auction is often used instead of a simpler, direct mechanism. Another example is the Dutch or descending auction.

The Dutch auction also consists of a number of rounds, except here it begins at a price p large enough that no bidders are interested, and successively decrements the price by ϵ until a bidder (or a set of bidders) declare their interest in the item. That bidder is declared the winner (or a tie is broken randomly); the winner receives the item and pays the final price reached by the auction.

Note that, the Dutch auction can be seen as the indirect mechanism analog of the first price auction, while the English auction can be seen as the indirect mechanism analog of the second price/Vickrey auction.

2 Auctions in Single-Parameter Environment

Before getting into multi-item auctions, we consider one more auction setting called the single parameter environments. Single parameter auctions are a generalization of single-item auctions to multiple copies of the item without assuming that bidders have a different value for each good. That is, in single parameter environments, we assume that the single private value that the bidder has is the same for all copies of the good. We now describe the model formally.

2.1 Model

The model consists of:

1. k copies of an indivisible good, i.e., the item can only be allocated in integer quantities.
2. A set of n bidders/buyers
3. Each bidder $i \in [n]$ has:
   - a private valuation v_i which corresponds to the maximum amount that bidder i is willing to pay for the auctioned good
   - a utility function $u_i(x_i; p_i) = x_i(v_i - p_i)$ where x_i is a binary variable denoting whether if the bidder i was allocated the good or not.

2.2 Outcome of the Model

A **single-parameter environment auction mechanism** $\mathcal{A} : \mathbb{R}^n \to \mathbb{N}^n \times \mathbb{R}$ is a function that takes as input the **reported valuations/bids** of the bidders and outputs a tuple consisting of an allocation vector $\boldsymbol{x} \in \mathbb{N}^n$ denoting the allocation of the auctioned good to i^{th} bidder i and a price $\boldsymbol{p} \in \mathbb{R}^n$. It is important to note that the bidders can report bids strategically, i.e., they can lie about their valuations to potentially improve the outcome for themselves.

Often, we will denote the allocation outputted by an auction mechanism $\mathcal{A}$ as a function of the bids, $\boldsymbol{b}$, by $\boldsymbol{x}(\boldsymbol{b})$, and the payment as a function $\boldsymbol{p}(\boldsymbol{b})$.

We can define indirect auction mechanisms in single-parameter environments as well, however, we focus in this section only on direct mechanisms. In particular, we present a revenue maximizing mechanism (which the auctions we have presented so far are not). The auction mechanisms we have provided in the previous section can be easily adapted to single parameter environments. Furthermore, not that the properties we have defined in the previous section are also valid in single parameter environments.

2.3 Designing Mechanisms in Single Parameter Environments

In this section, we will discuss important results for auction design in single parameter environments, then we will discuss how to design mechanism in single parameter environments:

Implementable Allocation Rule: An allocation rule $\boldsymbol{x}$ for a single-parameter environment is implementable if there is a payment rule $\boldsymbol{p}(\boldsymbol{b})$ such that the sealed-bid auction $(\boldsymbol{x}, \boldsymbol{p})$ is IC&IR. That is an allocation rule is implementable if there exists a payment rule that can enforce/incentivize that allocation (IC&IR allow us to enforce a certain allocation).

Monotone Allocation Rule: An allocation rule $\boldsymbol{x}(\boldsymbol{b})$ for a single-parameter environment is monotone if for every bidder i and bids $\mathbf{b}_{-i}$ by the other bidders, the allocation $x_i(b_i, \mathbf{b}_{-i})$ to bidder i is nondecreasing in its bid b_i.

We now introduce Myerson's lemma one of the most important results in mechanism design:

Theorem 2.1. *Myerson's lemma Fix a single-parameter environment.*

1. *An allocation rule $\boldsymbol{x}(\boldsymbol{b})$ is implementable if and only if it is monotone.*
2. *If $\boldsymbol{x}(\boldsymbol{b})$ is monotone, then there is a unique payment rule such that the sealed-bid mechanism $\mathcal{A}(\boldsymbol{b}) = (\boldsymbol{x}(\boldsymbol{b}), \boldsymbol{p}(\boldsymbol{b}))$ is IC&IR (assuming that when $b_i = 0$ then $\boldsymbol{p}(\boldsymbol{b}) = 0$)*
3. *The payment rule $\boldsymbol{p}(\boldsymbol{b})$ for the allocation rule $\boldsymbol{x}(\boldsymbol{b})$ is given by an explicit formula:*
   For a differentiable allocation rule $\boldsymbol{x}(\boldsymbol{b})$:

$$p_i(v_i, \mathbf{v}_{-i}) = v_i x_i(v_i, \mathbf{v}_{-i}) - \int_0^{v_i} x_i(b, \mathbf{v}_{-i})\, \mathrm{d}z \tag{1}$$

   For an allocation rule $\boldsymbol{x}(\boldsymbol{b})$ that is not differentiable or is discontinuous:

$$p_i(b_i, \mathbf{b}_{-i}) = \sum_{j=1}^{\ell} z_j \cdot \mathrm{jump\,in}\, x_i(\cdot, \mathbf{b}_{-i}) \ \textit{at}\ z_j \tag{2}$$

   where $z_1, \ldots, z_\ell$ are the breakpoints of the allocation function $x_i(\cdot, \mathbf{b}_{-i})$ in the range $[0, b_i]$

Myerson's lemma is an extremely important result because it allows us devise a simple recipe for mechanism design in single parameter environments which can be described as follows:

1. Assume that bids are truthful, pick an allocation rule $\boldsymbol{x}(\boldsymbol{b})$ such that it maximizes a desired objective function, i.e., $\boldsymbol{x}(\boldsymbol{b}) \in \arg\max_{\boldsymbol{x}} f(\boldsymbol{x})$
2. Devise the algorithm that calculates the allocation rule $\boldsymbol{x}(\boldsymbol{b})$ given the bids $\boldsymbol{b}$
3. Calculate the payment rule $\boldsymbol{p}(\boldsymbol{b})$ using the payment rule from Myserson's lemma

2.4 Revenue Maximizing Auctions in Single Parameter Environments with Priors

3 Multi-parameter auction

- VCG, Walrassian eqa. LP, clinching auction

4 Prophets and Simple Auction Mechanisms

- Prophets, reserve prices etc...

References