Mechanism Design

- Designing systems with strategic participants that have good performance guarantees

- Let's start with a simple case

- Single Item Auctions (1 item to sell)
  - n bidders (agents)
  - v_i: bidder $i's$ maximum willingness to pay (valuation) (private)
  - u_i: bidder $i's$ utility, defined as:

$$u_i = \begin{cases} 0 \\ v_i - p \end{cases}$$

Sealed Bid Auctions

- Each agent i submits bid b_i
- Using all bids, we want design a mechanism that decides
  - Allocation
  - Payment

- For example, one such mechanism could be give the item to the highest bidder and charge nothing
  - Is it a good mechanism?

everyone will bid infinity.

First Price Auction

$\max_{b_1} (v_1 - b_1) P(b_1 \text{ is biggest})$

- Allocation: give the item to the highest bidder
- Payment: the highest bid
- Is it a good mechanism?

$v_1 = 100$ $\quad v_2 = 50$ $\quad v_3 = 75$

$b_1 \cancel{=} 100$ $\quad b_2 \cancel{=} 50$ $\quad b_3 \cancel{=} 75$

20 $\quad$ 40 $\quad$ 45

want 76 → guess

difficult mechanism for bidders

also not good for autioner, because he doesn't know whether the item is given to the one who values the most.

Second Price Auction

- Allocation: give the item to the highest bidder
- Payment: the second highest bid

- Is it a good mechanism?

$v_1 = 100$ $v_2 = 50$ $v_3 = 75$

$b_1 = 100$ $b_2 = 50$ $b_3 = 75$

it is best for every bidder to bid their true valuations.

item will go to the agent who values the most.

- "proxy bidder" used by eBay

autioner will also know bidder's true valuations.

Second Price Auction (Vickery Auction)

Claim: Every bidder has a dominant strategy where $b_i = v_i$, i.e., this strategy maximizes the utility regardless of what other bidders do

$B = \max_{j \neq i} \{ b_j \}$.

Proof:

Case 1: $B > V_i$. what is the best response of bidder i?

if $b_i > B$ win $\Rightarrow u = V_i - B < 0$

if $b_i \le B$ loss $\Rightarrow u = 0$ best in case 1.

$\rightarrow b_i = V_i$.

Case 2: $B \le V_i$

if $b_i > B$ win $\Rightarrow u = V_i - B \ge 0$ best in case 2

if $b_i \le B$ lose $\Rightarrow u = 0$.

Second Price Auction (Vickery Auction)

Claim: Every truth-telling bidder is guaranteed non-negative utility

Proof: $b_i = v_i$.

$$u_i = \begin{cases} 0 & \text{if } i \text{ doesn't get the item.} \\ v_i - p & \text{if } i \text{ gets the item.} \end{cases}$$

Since $p \le b_i = v_i \Rightarrow u_i \geq 0 \quad \forall i.$

Second Price Auction (Vickery Auction)

- Vickery Auction is awesome: It has remarkable properties
  - Dominant Strategy Incentive Compatible (DSIC): Acting according to the true preferences is the best strategy regardless of what others do

x_i : allocation of agent i.

  - Maximizes Social Welfare ← item is given to the agent who values the most.

$\max \sum v_i x_i$.

  - Computational Efficiency

maximum of n numbers → $O(n)$ $\quad O(n) + O(n) = O(n)$

payment → second maximum → $O(n)$.

  - In addition, auctioneer know the valuations (private info) of all bidders!

- We want to design a mechanism with such remarkable properties beyond single item auction

Sponsored Search

- k slots; each slot has a click-through-rate (CTR), say
$$\alpha_1 \geq \alpha_2 \geq \cdots \geq \alpha_k$$
- Mechanism needs to decide allocation (x) and payment (p)

- Impression vs. Click
  - Assume that advertisers are not interested in impression, but on clicks

- v_i: valuation of bidder i for a click
- If j^{th} slot is allocated to bidder i then $i's$ value is = ? $\alpha_j v_i$.

Mechanism Design

- We want mechanism to have remarkable properties
  1) DSIC: truthful bidding is a dominant strategy $\longrightarrow b_i = v_i$.
  2) Social welfare maximization: X = set of all possible allocations
  3) Fast running time (polynomial) $= \{ (\alpha_1, \alpha_2, \dots, \alpha_k, 0, \dots, 0),$

$(\alpha_2, \alpha, \dots$

$\dots$

$(0, 0, \dots 0, \alpha_1, \dots, \alpha_k) \}.$

$$\max_{x \in X} \sum_j v_j x_j.$$

Mechanism Design

- We want mechanism to have remarkable properties
  1) DSIC: truthful bidding is a dominant strategy
  2) Social welfare maximization:
  3) Fast running time (polynomial)

Design Approach

- Step 1: First assume that bidders bid truthfully. How should be assign bidders to slots so that (2) and (3) holds?
- Step 2: Given the answer of Step 1, how should we set selling prices so that (1) holds?

How to Implement Steps 1 and 2?

$k = 3 \quad n = 5.$

- Step 1: $b = 15 \quad 20 \quad 25 \quad 10 \quad 35$

  $x(b) = (0 \quad \alpha_3 \quad \alpha_2 \quad 0 \quad \alpha_1).$

  computationally effective.

- Step 2: Are there prices that render truthful bidding a dominant strategy for every bidder?

If $\alpha_1 = \alpha_2 = \cdots = \alpha_k = 1 \rightarrow p(b) = \underline{(k+1)^{th} \text{ highest bid}} \longleftarrow$ DSIC.

Myerson's Lemma

- Single Parameter Setting
  - n bidders
  - Each bidder i has valuation v_i per unit of allocation
  - X: feasible set of all allocation, where each element of X is $(x_1, \ldots, x_n)$ so that x_i is the amount given to bidder i

- Examples
  - Single Item Auction
  - k Identical Goods where each bidder can get at most one good
  - Sponsored Search where each bidder can get at most one slot and each slot can be given to at most one bidder
    - If i is assigned slot j, then $x_i =?$ α_j.

Allocation & Payment Rule (x, p)

- Bids $b = (b_1, \ldots, b_n)$
- Allocation rule $x(b) \in X$
- Payment rule $p(b) \in R^n$
- What is $u_i(b) = ?$ $x_i(b) v_i - p_i(b)$.

- We want that $p_i(b) \in [0, b_i . x_i(b)]$ for every i and b

Implementable Allocation Rule

- An allocation x for a $\boxed{\text{single-parameter environment}}$ is implementable if there is a payment rule $p(.)$ such that the sealed bid auction (x, p) is DSIC

- Is the welfare maximization rule for sponsored search, which assigns j^{th} highest slot to j^{th} highest bidder implementable?

- Example: In single item auction, if we give the item to, say the second highest bidder, then is there a payment rule to make it DSIC?

Monotone Allocation Rule

- An allocation x for a single-parameter environment is monotone if for every bidder i and bids b_{-i} of other bidders, the allocation $x_i(a, b_{-i})$ to i is nondecreasing in its bid a

- Examples:
  - Single Item Auction
  - Sponsored Search

Myerson's Lemma

- For a single parameter environment
  - An allocation rule x is implementable if and only if it is monotone
  - If x is monotone, then there exists a unique payment rule such that the sealed-bid mechanism (x, p) is DISC [assuming that $b_i = 0 \Rightarrow p_i(b) = 0$ (normalization)]
  - The payment rule is given by an explicit formula

- There is no ambiguity in how to assign payments to achieve the DISC property!

Proof of Myerson's Lemma

- x: allocation rule (need not be monotone)
- p: payment rule

Let's apply the DSIC property

- Fix i and b_{-i} arbitrarily
- $x(a) := x_i(a, b_{-i})$ and $p(a) \coloneqq p_i(a, b_{-i})$
- Consider $0 \leq \alpha < \beta$
- i might have private valuation β and can submit α
- DSIC demands that: utility of bidding $\beta \geq$ utility of bidding α

Proof: Consider $0 \leq \alpha < \beta$

i might have private valuation β and can submit α

DSIC demands that: utility of bidding $\beta \geq$ utility of bidding α

① x is Implementable $\Longrightarrow$ (x,p) is DSIC $\Longrightarrow$ (x,p) satifies:

$$v\, x_i(v, b_{-i}) - p_i(v, b_{-i}) \geqslant v\, x_i(v', b_{-i}) - p_i(v', b_{-i}).$$

$$v'\, x_i(v', b_{-i}) - p_i(v', b_{-i}) \geqslant v'\, x_i(v, b_{-i}) - p_i(v, b_{-i}).$$

$$\Rightarrow v\,(x_i(v, b_{-i}) - x(v', b_{-i})) \geqslant p_i(v, b_{-i}) - p_i(v', b_{-i}).$$

$$v'\,(x_i(v, b_{-i}) - x(v', b_{-i})) \leq p_i(v, b_{-i}) - p_i(v', b_{-i})$$

$$\Rightarrow v\,(x_i(v, b_{-i}) - x_i(v', b_{-i})) \geqslant v'\,(x_i(v, b_{-i}) - x(v', b_{-i}))$$

$\Rightarrow$ x is monotone.

We are building a mechanism for general situation, so the i's value could by any constants. We consider two different situation i's value is v and v' this place.

Take $v' = v - \varepsilon$, $\varepsilon \to 0$

$$v\,(x_i(v, b_{-i}) - x_i(v', b_{-i})) - \varepsilon\,(x_i(v, b_{-i}) - x_i(v', b_{-i}))$$

$$\leq p_i(v, b_{-i}) - p_i(v', b_{-i}) \leq v\,(x_i(v, b_{-i}) - x_i(v', b_{-i}))$$

$$[p_i(v, b_{-i}) - \cancel{p_i(v-\varepsilon, b_{-i})}] + [\cancel{p_i(v-\varepsilon, b_{-i})} - \cancel{p_i(v-2\varepsilon, b_{-i})}]$$

$$+ \cdots + [\cancel{p_i(\varepsilon, b_{-i})} - p_i(0, b_{-i})].$$

$$v\,(x_i(v, b_{-i}) - x_i(v-\varepsilon, b_{-i})) + (v-\varepsilon)(x_i(v-\varepsilon, b_{-i}) - x_i(v-2\varepsilon, b_{-i}))$$

$$+ \cdots + \varepsilon\,(x_i(\varepsilon, b_{-i}) - x_i(0, b_{-i})).$$

$$= v\, x_i(v, b_{-i}) - \varepsilon\left[x_i(v-\varepsilon, b_{-i}) + \cdots x_i(\varepsilon, b_{-i}) + x_i(0, b_{-i})\right]$$

$$\leq \varepsilon\left(x_i(v, b_{-i}) - \cancel{x_i(v-\varepsilon, b_{-i})}\right) + \varepsilon\left(\cancel{x_i(v-\varepsilon, b_{-i})} - \cancel{x_i(v-2\varepsilon, b_{-i})}\right) + \cdots\cdots + \varepsilon\left(\cancel{x_i(\varepsilon, b_{-i})} - x_i(0, b_{-i})\right)$$

$$= \varepsilon\left(x_i(v, b_{-i}) - x_i(0, b_{-i})\right).$$

$$\Rightarrow v\, x_i(v, b_{-i}) - \varepsilon\left[x_i(v, b_{-i}) + \cdots + x_i(\varepsilon, b_{-i})\right].$$

$$\leq p_i(v, b_{-i}) - p_i(0, b_{-i})$$

$$\leq v\, x_i(v, b_{-i}) - \varepsilon\left[x_i(v-\varepsilon, b_{-i}) + \cdots\cdots + x_i(0, b_{-i})\right].$$

As $\varepsilon \to 0$

$$p_i(v, b_{-i}) - p_i(0, b_{-i})$$

$$= v\, x_i(v, b_{-i}) - \int_0^v x_i(z, b_{-i})\,dz$$

$$\Rightarrow p_i(v, b_{-i}) = v\, x_i(v, b_{-i}) - \int_0^v x_i(z, b_{-i})\,dz$$

$$+ \underline{p_i(0, b_{-i})}$$ could set to 0.

$$p_i(v, b_{-i}) = v\, x_i(v, b_{-i}) - \int_0^v x_i(z, b_{-i})\,dz$$

② monotone $\Rightarrow$ implementable

i.e. prove $p = \{ p_i(v, b_{-i}) = v\, x_i(v, b_{-i}) - \int_0^v x_i(z, b_{-i})\,dz + p_i(0, b_{-i}) \}$.

(x, p) is DSIC.

Set $v > v'$.

$v\, x_i(v, b_{-i}) - p_i(v, b_{-i}) = \int_0^v x_i(z, b_{-i})\,dz - p_i(0, b_{-i})$.

$v\, x_i(v', b_{-i}) - p_i(v', b_{-i})$

$= (v - v')\, x_i(v', b_{-i}) + \int_0^{v'} x_i(z, b_{-i})\,dz - p_i(0, b_{-i})$.

$\Rightarrow [v\, x_i(v', b_{-i}) - p_i(v', b_{-i})] - [v\, x_i(v, b_{-i}) - p_i(v, b_{-i})]$

$= (v - v')\, x_i(v', b_{-i}) - \int_{v'}^{v} x_i(z, b_{-i})\,dz$

(Monotone $\Rightarrow \forall z \in [v', v]\ \ x_i(z, b_{-i}) \geq x_i(v', b_{-i})$.)

$\Rightarrow v\, x_i(v', b_{-i}) - p_i(v', b_{-i}) \leq v\, x_i(v, b_{-i}) - p_i(v, b_{-i})$.

Similiarly, if $v' > v$.

$[v\, x_i(v', b_{-i}) - p_i(v', b_{-i})] - [v\, x_i(v, b_{-i}) - p_i(v, b_{-i})]$

$= -(v' - v)\, x_i(v', b_{-i}) + \int_{v}^{v'} x_i(z, b_{-i})\,dz$

(Monotone $\Rightarrow \forall z \in [v, v']\ \ x_i(z, b_{-i}) \leq x_i(v', b_{-i})$.)

$\Rightarrow$ DSIC.