Mechanism Design (Contd.)

Tuesday, September 20, 2022 10:58 AM

Last lec: Single parameter (single stuff)

agent i has value V_i/unit-stuff

bids b_i

$$\bar{b} = (b_1, \dots, b_n) \rightarrow \boxed{\text{Mechanism}} \rightarrow \begin{array}{l} x_i(\bar{b}) \quad \forall i \\ P_i(\bar{b}) \end{array}$$

$$U_i(\bar{b}) = V_i \cdot x_i(\bar{b}) - P_i(\bar{b})$$

★ Myerson's lemma:

① (x, P) is DSIC $\Leftrightarrow$ $x(\cdot)$ is monotone

$\hookrightarrow$ $\forall i, \forall b_{-i}$, $x_i(b_i, b_{-i})$ wrt b_i is monotone.

② $\exists$ unique P s.t. (x, P) is DSIC

③ P has an explicit formula.

Pf:

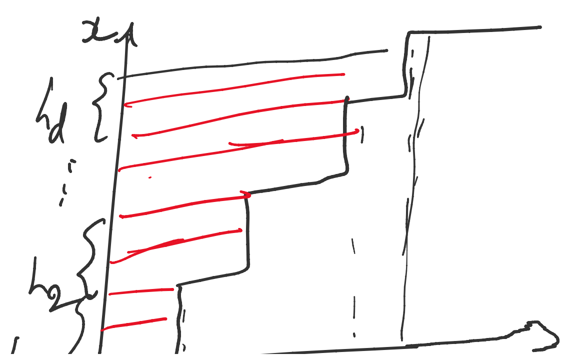
claim 1: (x, P) is DSIC $\Rightarrow$ $x(\cdot)$ is monotone.

claim 2: If (x, P) DSIC, then P has an explicit formula.

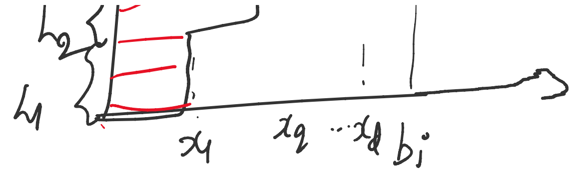
Fix i, Fix b_{-i}

$x(b_i) = x_i(b_i, b_{-i})$

$p(b_i) = P_i(b_i, b_{-i})$

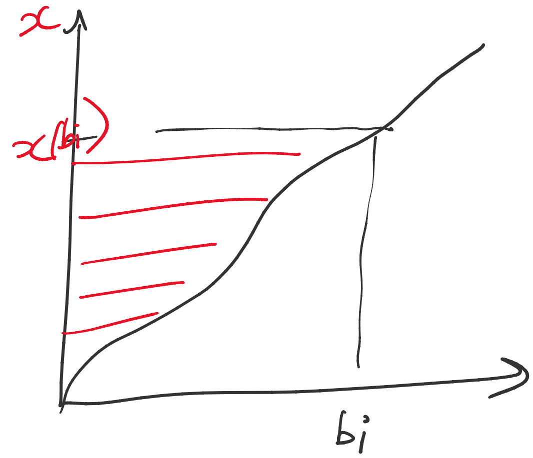


$$p(b_i) = P_i(b_i, b_{-i})$$



$$P(b_i) = \sum_{k=1}^{d} x_d \cdot h_d$$

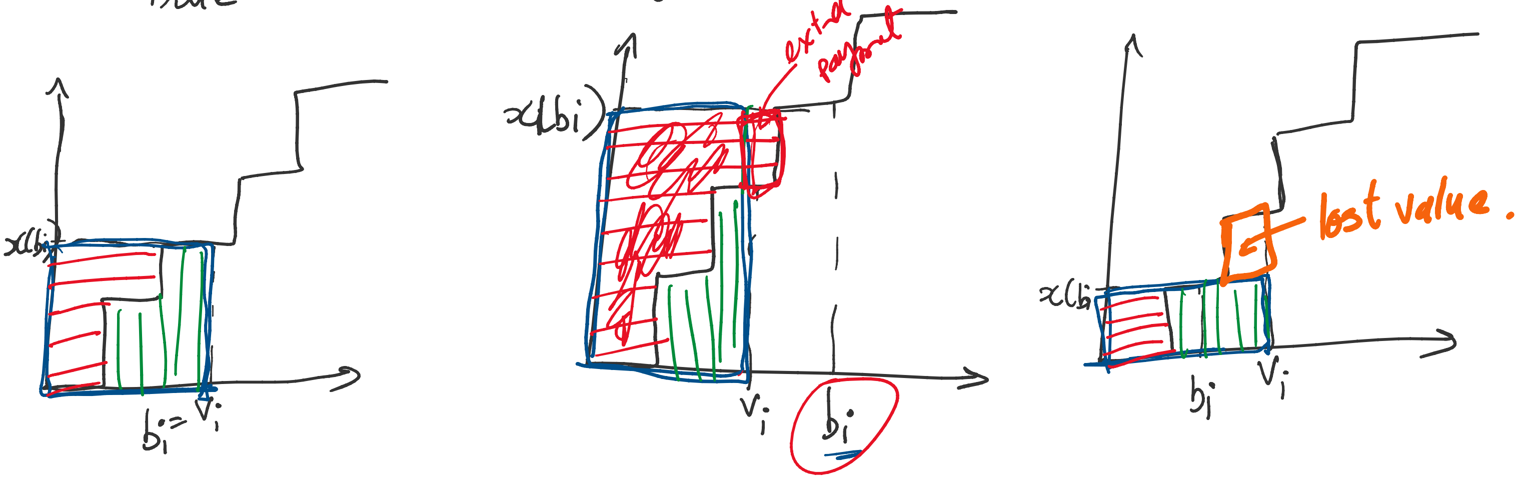
$$P(b_i) = \int_0^{b_i} z\, x'(z)\, dz$$



Claim 3: Such (x, P) is DSIC

Pf: Fix i, b_{-i} $\qquad U_i = v_i \cdot x(b_i) - P_i$

True $\qquad$ over bid $\qquad$ underbid



★ Awesome Auctions.

① DSIC ② S.W. maximizing ③ efficient implementation

↳ Fixes $x(\cdot) = \underset{x \in X}{\arg\max} \sum_{i \in N} b_i x_i$

Monotone? YES! (exe).

Monotone? YES! (exe).

↳ (Day 1) Fixes $p(\cdot)$ (Myerson's Lemma).

$\Downarrow$

Awesome Auction $\xrightarrow{\text{reduces}}$ Design "Algorithm" to find s.w. maximizing allocation.

↳ efficient (poly-time)?

Eg: TV Ad. Auction.

W sec of commercial break.

N: set of advertiser.

$i \in N$: ① ad of length w_i secs.

② value v_i (private)

③ bid b_i

→ set of allocations

$$X = \left\{ x \in \{0,1\}^N \;\middle|\; \sum_{i\in N} x_i w_i \le W \right\}.$$

→ S.W. max allocation

$$x(\cdot) = \arg\max_{x \in X} \sum_{i \in N} x_i b_i.$$

← Knapsack.

NP-hard.

* Greedy Algo.

① $\frac{b_1}{w_1} > \frac{b_2}{w_2} > \dots > \frac{b_n}{w_n}$

allocate in this order until space allows.

Suppose allocate $1, \dots, k$

② $i^* \in \arg\max_i b_i$

If $\sum_{i=1}^{k} x_i b_i > b_{i^*}$ the allocate $1 \dots k$

Else allocate only i^*.

Thm: Greedy s.w. $\geq \frac{1}{2}$ max. s.w.

pf: (exe).

claim: Greedy allocation rule is monotone!

$\Downarrow$

$\frac{1}{2}$ Awesome auction:
- DSIC
- $1/2$ max s.w.
- efficiently implementable

* $(1+\epsilon)$-approx. algo for knapsack

$\hookrightarrow$ $x(\cdot)$ is not monotone!

Can we make it monotone?

* Many other setting give rise to NP-hard opt problems ...

★ Many other setting give rise to NP-hard opt.

→ auctions facilities

→ auction machine time

→ Spectrum auction.

$\boxed{\alpha\text{-approx algo for max s.w.}} \xrightarrow{\text{give}} \boxed{\alpha\text{-awesome auction} \equiv \text{DSIC mechanism?}}$

↓

YES, if corresponding $x(\cdot)$ is monotone.

But $x(\cdot)$ need not be monotone.

Can be made monotone if X is downward closed.

if $x \in X$ then $\forall y \leq x$, $y \in X$.

(dawla et al.)

General setting : Multi parameter.

e.g. auction of k-heterogeneous items

(painting, laptop, phone, pen, ...)

→ Ω : set of possible outcomes

Single-item.

$\Omega = \{i\text{-wins} \mid i \in N\}$

→ Ω: set of possible outcomes

→ For every agent $i \in N$

① $v_i : \Omega \to \mathbb{R}$ (private)

② $b_i : \Omega \to \mathbb{R}$ (bid).

$\Omega = \{i\text{-wins} \mid i \in N\}$

$v_i(i\text{-wins}) = v_i$

$v_i(j\text{-wins}) = 0 \quad \forall j \neq i$

Goal: Design a DSIC Mechanism.

$(b_1, \ldots, b_n) \to$ Mechanism $\to \omega^* \in \Omega$, $\to p_i : \forall i$

functions

$$\boxed{U_i = v_i(\omega^*) - p_i}$$

★ Vickery-Clark-Groove (VCG) Mechanism:
[only DSIC Mechanism].

① $\omega^* \in \underset{\omega \in \Omega}{\text{argmax}} \sum_{i \in N} b_i(\omega)$

"pain" = "loss of value"

② $\forall i$, p_i = "externality" agent i causes to the s.w. of everyone else by participating in the auction.

$= \left[\max \sum b_k(\omega) \right] - \sum b_k(\omega^*)$

$$= \boxed{\underbrace{\max_{\omega \in \Omega} \sum_{k \neq i} b_k(\omega)}_{\text{s.w. of others when } i \text{ does not participate}}} - \underbrace{\sum_{k \neq i} b_k(\omega^*)}_{\text{s.w. of others when } i \text{ participates}}$$

independent of b_i

$h_i(b_{-i})$

Thm: VCG is DSIC.

Pf:

$$U_i = v_i(\omega^*) - p_i$$

$$= v_i(\omega^*) - \left(\underline{h_i(b_{-i})} - \sum_{k \neq i} b_k(\omega^*) \right)$$

$$= \boxed{v_i(\omega^*) + \sum_{k \neq i} b_k(\omega^*)} - h_i(b_{-i})$$

$\to$ i wants maximize this quantity.

Thought exp: suppose we allow i to pick the outcome.

i pick : $\arg\max_{\omega \in \Omega} v_i(\omega) + \sum_{k \neq i} b_k(\omega)$ $\to$

same if $b_i = v_i$

auctioneer picking : $\arg\max_{\omega \in \Omega} b_i(\omega) + \sum_{k \neq i} b_k(\omega)$ $\to$

i achieves the goal by bidding $b_i = v_i$

(truthfully).

i achieves the goals of

(truthfully).

$\Downarrow$

VCG is DSIC ∎.

Issues:

1. Implementation: efficient?
2. Representation of b_i

   10 item

3. Revenue.
4. Signaling (indirect implementation).

Q: Is VCG same as Myerson for single-parameter?

e.g. single item, k-identical items, sponsored search?