Lecture 18

Monday, 7 November 2022 12:05 PM

(first part is in Lecture 17)

Multi - Unit & Knapsack Auctions

Multi - unit auctions :

k identical items to be auctioned

so : $A = \{ x \in \{0,1\}^n : \sum_i x_i \leq k \}$

Single - parameter domain, so $\forall i$, $v \in \mathbb{R}_+$, $b_i \in \mathbb{R}_+$

(thus, $V_i = B_i = \mathbb{R}_+$)

SCF : assign items to maximize SW (i.e, assign items to agents w/ highest v_i)

Problem : design a DSIC mechanism for this SCF. $M = (x, p)$

Design Paradigm :

Step 1: Assume truthful bids. Choose $\overset{\text{monotone}}{x : \mathbb{R}_+^n \to A}$ to attain objective

Step 2: Use Myerson's Lemma to compute prices s.t. $M = (x, p)$ is IC (and hence bids are indeed truthful)

Assume all v_i's are distinct

Given bids $b_1, \ldots, b_n$, let $\pi \in S_n$ be a permutation s.t. $\pi(i)$ is the i^{th} highest bidder.

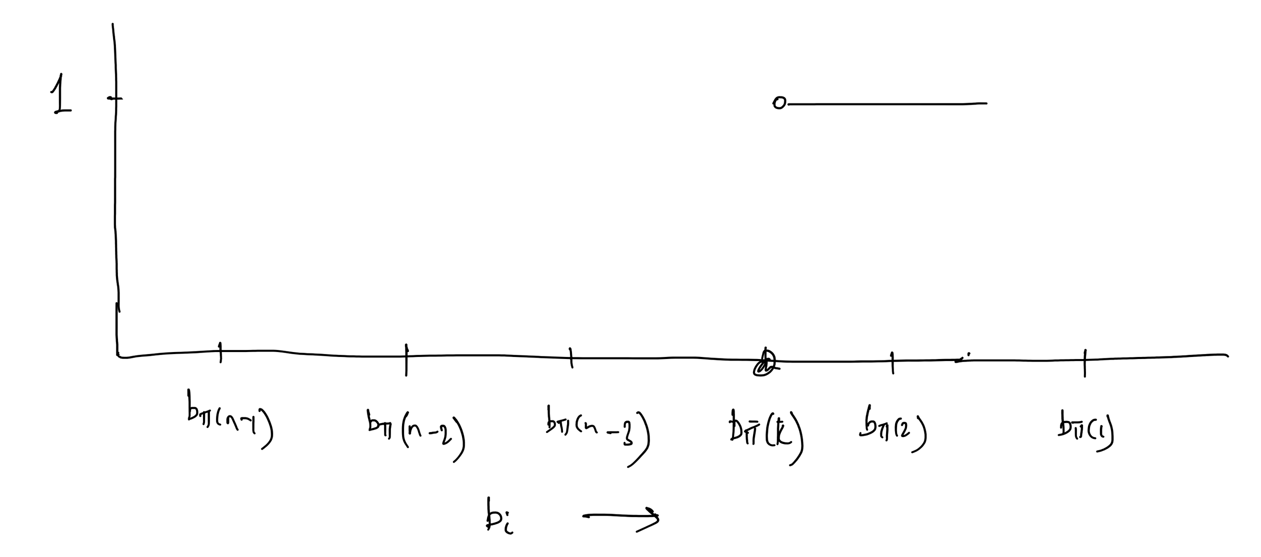
Then $b_{\pi(1)} \geq b_{\pi(2)} \geq \ldots \geq b_{\pi(n)}$

Give items to k highest bidders, i.e,

$$x_i = \begin{cases} 1 & \text{if } \pi^{-1}(i) \in [k] \\ 0 & \text{o.w.} \end{cases}$$

Clearly, x_i is monotone.

Payment $p_i(b_i, b_{-i})$



(bids of other $(n-1)$ bidders) :

Thus if $b_i > b_{\pi(k+1)}$, $x_i(b) = 1$ & $p_i(b) = b_{\pi(k+1)}$

Thus, $b_{\pi(k+1)}$ is the "critical bid" for each bidder.

Knapsack Auctions

- Knapsack of size S
- n bidders, each has an item of (private) value v_i & public size $s_i \leq S$
- $A = \{ x \in \{0,1\}^n : \sum_i s_i x_i \leq S \}$
- SCF f: maximize value of items put into knapsack, subject to size constraints

$M = (x, p)$: as per design Paradigm:

Step 1: $x(b) \in \arg\max_{y \in A} \sum_i b_i y_i$

(is this monotone ?)

Step 2: compute payments ...

Problem is, step 1 requires us to solve an NP- complete problem !

Even if we require / assume players tell the truth (i.e., relax DSIC), computational constraints are still an issue. for computing the SCF itself.

Instead we try to "approximately" maximize the SW.

Defn : Approx ratio : For a maximization problem, algorithm A has an approx ratio of α if for every instance I, $\frac{ALG(I)}{OPT(I)} \geq \alpha$

For the knapsack <u>problem</u> (i.e., given (s_i, v_i) $\forall i \in [n]$), the following is a well - known $1/2$ - approx algo :

Knapsack Algo :

- order items s.t. $\frac{v_1}{s_1} \geq \frac{v_2}{s_2} \geq \ldots \geq \frac{v_n}{s_n}$
- Starting with the first item, put items into knapsack until it is full.
- return Either this solution, or the highest value item, whichever has higher total value.

Claim: The above algo is a $1/2$ - approx algo for knapsack.

Proof: Let $x^A \in \{0,1\}^n$ be the soln, our algo returns, while x^o is the optimal soln. Further, let $x^f \in [0,1]^n$ be the optimal fractional soln. , i.e, the soln. to

$\max \sum_i x_i v_i$ s.t. $\sum_i x_i s_i \leq S$, $\forall i$ $x_i \in [0,1]$

Note that : (I) $\sum_i v_i x_i^A \leq \sum_i v_i x_i^o \leq \sum_i v_i x_i^f$, &

(II) with items indexed by the ratio above, $x_1^f = \ldots = x_k^f = 1$, $x_{k+1}^f \in [0,1)$, and $x_i^f = 0$ $\forall i > k+1$

We will show that $\sum_i x_i^A v_i \geq \frac{1}{2} \sum_i v_i x_i^f$.

Consider 2 cases:

(I) $v_{k+1} \geq \sum_{i \leq k} v_i$

Then $\sum_i x_i v_i^A \geq \max_i v_i \geq v_{k+1}$

$\geq \frac{1}{2} \left(v_{k+1} + \sum_{i \leq k} v_i \right)$

$\geq \frac{1}{2} \sum_i x_i v_i^f$

(II) $v_{k+1} < \sum_{i \leq k} v_i$

Then $\sum_i x_i v_i^A \geq \sum_{i \leq k} v_i \geq \frac{1}{2} \left(\sum_{i \leq k} v_i + v_{k+1} \right)$

$\geq \frac{1}{2} \sum_i x_i v_i^f$

Thus, in either case, the claim holds.

Claim : The allocation x^A is monotone in v_i's

(show yourself)

Since the algorithm is monotone, & obtains at least the maximum social welfare, we can design a payment fn. to convert this into a DSIC mechanism that obtains at least $1/2$ the social welfare.