Combinatorial Auctions

Lecture 21

Lecture Overview

Designing optimal auctions

Definition (virtual valuation)

Bidder i's virtual valuation is $\psi_i(v_i) = v_i - \frac{1-F_i(v_i)}{f_i(v_i)}$.

Definition (bidder-specific reserve price)

Bidder i's bidder-specific reserve price r_i^* is the value for which $\psi_i(r_i^*) = 0$.

Theorem

The optimal (single-good) auction is a sealed-bid auction in which every agent is asked to declare his valuation. The good is sold to the agent $i = \arg\max_i \psi_i(\hat{v}_i)$, as long as $v_i > r_i^*$. If the good is sold, the winning agent i is charged the smallest valuation that he could have declared while still remaining the winner: $\inf\{v_i^* : \psi_i(v_i^*) \geq 0$ *and* $\forall j \neq i,\ \psi_i(v_i^*) \geq \psi_j(\hat{v}_j)\}$.

Analyzing optimal auctions

Optimal Auction:

- winning agent: $i = \arg\max_i \psi_i(\hat{v}_i)$, as long as $v_i > r_i^*$.
- i is charged the smallest valuation that he could have declared while still remaining the winner,
  $\inf\{v_i^* : \psi_i(v_i^*) \geq 0$ and $\forall j \neq i,\ \psi_i(v_i^*) \geq \psi_j(\hat{v}_j)\}$.

- it's a second-price auction with a reserve price, held in virtual valuation space.
- neither the reserve prices nor the virtual valuation transformation depends on the agent's declaration
- thus the proof that a second-price auction is dominant-strategy truthful applies here as well.

Going beyond IPV

- common value model
  - motivation: oil well
  - winner's curse
  - things can be improved by revealing more information
- general model
  - IPV + common value
  - example motivation: private value plus resale

Risk Attitudes

What kind of auction would the auctioneer prefer?

- Buyer is not risk neutral:
  - no change under various risk attitudes for second price
  - in first-price, increasing bid amount increases probability of winning, decreases profit. This is good for risk-averse bidder, bad for risk-seeking bidder.
  - Risk averse, IPV: First $\succ$ [Japanese $=$ English $=$ Second]
  - Risk seeking, IPV: Second $\succ$ First
- Auctioneer is not risk neutral:
  - revenue is fixed in first-price auction (the expected amount of the second-highest bid)
  - revenue varies in second-price auction, with the same expected value
  - thus, a risk-averse seller prefers first-price to second-price.

Multiunit Auctions

- now let's consider a setting in which
  - there are k identical goods for sale in a single auction
  - every bidder only wants one unit
- VCG in this setting:
  - every unit is sold for the amount of the $k+1$st highest bid
- revenue equivalence holds here, so all other methods of setting prices lead to the same payments in equilibrium.

Lecture Overview

1. Recap
2. General Multiunit Auctions
3. Combinatorial Auctions
4. Bidding Languages

Multiunit Demand

How does VCG behave when (some) bidders may want more than a single unit of the good?

Multiunit Demand

How does VCG behave when (some) bidders may want more than a single unit of the good?

- no longer a $k+1$st-price auction
- instead, all winning bidders who won the same number of units will pay the same amount as each other.
  - the change in social welfare from dropping any of these bidders is the same.
- Bidders who win different numbers of units will not necessarily pay the same per unit prices.
- However, bidders who win larger numbers of units will pay at least as much in total (not necessarily per unit) as bidders who won smaller numbers of units
  - their impact on social welfare will always be at least as great

Winner Determination for Multiunit Demand

- Let m be the number of units available, and let $\hat{v}_i(k)$ denote bidder i's declared valuation for being awarded k units.
- It's no longer computationally easy to identify the winners—now it's a (NP-complete) weighted knapsack problem:

$$\text{maximize} \quad \sum_{i \in N} \sum_{1 \leq k \leq m} \hat{v}_i(k) x_{k,i} \tag{1}$$

$$\text{subject to} \quad \sum_{i \in N} \sum_{1 \leq k \leq m} k \cdot x_{k,i} \leq m \tag{2}$$

$$\sum_{1 \leq k \leq m} x_{k,i} \leq 1 \qquad \forall i \in N \tag{3}$$

$$x_{k,i} = \{0, 1\} \qquad \forall 1 \leq k \leq m, i \in N \tag{4}$$

Winner Determination for Multiunit Demand

$$\text{maximize} \quad \sum_{i \in N} \sum_{1 \le k \le m} \hat{v}_i(k) x_{k,i} \tag{1}$$

$$\text{subject to} \quad \sum_{i \in N} \sum_{1 \le k \le m} k \cdot x_{k,i} \le m \tag{2}$$

$$\sum_{1 \le k \le m} x_{k,i} \le 1 \qquad \forall i \in N \tag{3}$$

$$x_{k,i} = \{0, 1\} \qquad \forall 1 \le k \le m, i \in N \tag{4}$$

- $x_{k,i}$ indicates whether bidder i is allocated exactly k units
- maximize: sum of agents' valuations for the chosen allocation
- (2): number of units allocated does not exceed number available
- (3): no more than one $x_{\cdot,i}$ is nonzero for any i
- (4): all x's must be integers

Multiunit Valuations

How can bidders express their valuations in a multiunit auction?

- m homogeneous goods, let S denote some set
- general: let $p_1, \ldots, p_m$ be arbitrary, non-negative real numbers. Then $v(S) = \sum_{j=1}^{|S|} p_j$.
- downward sloping: general, but $p_1 \geq p_2 \geq \ldots \geq p_m$
- additive: $v(S) = c|S|$
- single-item: $v(S) = c$ if $s \neq \emptyset$; 0 otherwise
- fixed-budget: $v(S) = \min(c|S|, b)$
- majority: $v(S) = c$ if $|S| \geq m/2$, 0 otherwise

Advanced Multiunit Auctions

- Unlimited supply: random sampling auctions
  - how to sell goods that cost nothing to produce, when the valuation distribution is unknown?
- Search engine advertising: position auctions
  - how to sell slots on the right-hand side of internet search results

Lecture Overview

Valuations for heterogeneous goods

- now consider a case where multiple, heterogeneous goods are being sold.
- consider the sorts of valuations that agents could have in this case:
  - complementarity: for sets S and T, $v(S \cup T) > v(S) + v(T)$
    - e.g., a left shoe and a right shoe
  - substitutability: $v(S \cup T) < v(S) + v(T)$
    - e.g., two tickets to different movies playing at the same time
- substitutability is relatively easy to deal with
  - e.g., just sell the goods sequentially, or allow bid withdrawal
- complementarity is trickier...

Fun Game

1	2	3
4	5	6
7	8	9

- 9 plots of land for sale, geographically related as shown
- IPV, normally distributed with mean 50, stdev 5
- payoff:
  - if you get one good other than #5: v_i
  - any two goods: $3v_i$
  - any three (or more) goods: $5v_i$
- Rules:
  - auctioneer moves from one good to the next sequentially, holding an English auction for each good.
  - bidding stops on a good: move on to the next good
  - no bids for any of the 9 goods: end the auction

Combinatorial auctions

- running a simultaneous ascending auction is inefficient
  - exposure problem
  - inefficiency due to fear of exposure
- if we want an efficient outcome, why not just run VCG?
  - unfortunately, it again requires solving an NP-complete problem
  - let there be n goods, m bids, sets C_j of XOR bids
  - weighted set packing problem:

$$\max \sum_{i=1}^{m} x_i p_i$$

$$\text{subject to} \sum_{i|g \in S_i} x_i \leq 1 \qquad \forall g$$

$$x_i \in \{0, 1\} \qquad \forall i$$

$$\sum_{k \in C_j} x_k \leq 1 \qquad \forall j$$

Combinatorial auctions

$$\max \sum_{i=1}^{m} x_i p_i$$

$$\text{subject to} \sum_{i|g\in S_i} x_i \leq 1 \qquad \forall g$$

$$x_i \in \{0,1\} \qquad \forall i$$

$$\sum_{k\in C_j} x_k \leq 1 \qquad \forall j$$

- we don't need the XOR constraints
  - instead, we can introduce "dummy goods" that don't correspond to goods in the auction, but that enforce XOR constraints.
  - amounts to exactly the same thing: the first constraint has the same form as the third

Winner determination problem

How do we deal with the computational complexity of the winner determination problem?

- Require bids to come from a restricted set, guaranteeing that the WDP can be solved in polynomial time
  - problem: these restricted sets are *very* restricted...
- Use heuristic methods to solve the problem
  - this works pretty well in practice, making it possible to solve WDPs with many hundreds of goods and thousands of bids.

Lecture Overview

1. Recap
2. General Multiunit Auctions
3. Combinatorial Auctions
4. Bidding Languages

Expressing a bid in combinatorial auctions: OR bidding

- Atomic bid: (S, p) means $v(S) = p$
  - implicitly, an "AND" of the singletons in S
- OR bid: combine atomic bids
- let v_1, v_2 be arbitrary valuations

$$(v_1 \vee v_2)(S) = \max_{\substack{R,T \subseteq S \\ R \cup T = \emptyset}} [v_1(R) + v_2(S)]$$

Theorem

OR bids can express all valuations that do not have any substitutability, and only these valuations.

XOR Bids

- XOR bidding: allow substitutabilities
  - $(v_1 XOR v_2)(S) = \max(v_1(S), v_2(S))$

Theorem

XOR bids can represent any valuation

- this isn't really surprising, since we can enumerate valuations
- however, this implies that they don't represent everything efficiently

Theorem

Additive valuations require linear space with OR, exponential space with XOR

- likewise with many other valuations: any in which the price is different for every bundle

Composite Bidding Languages

- OR-of-XOR
- sets of XOR bids, where the bidder is willing to get either one or zero from each set
  - $(\ldots XOR \ldots XOR \ldots) OR (\ldots) OR (\ldots)$

Theorem

Any downward sloping valuation can be represented using the OR-of-XOR language using at most m^2 atomic bids.

- XOR-of-OR
  - a set of OR atomic bids, where the bidder is willing to select from only one of these sets
- generalized OR/XOR
  - arbitrary nesting of OR and XOR

The OR* Language

- OR*
  - OR, but uses dummy goods to simulate XOR constraints

Theorem

OR-of-XOR size $k \Rightarrow$ *OR* size* k, $\leq k$ *dummy goods*

Theorem

Generalized OR/XOR size $k \Rightarrow$ *OR* size* k, $\leq k^2$ *dummy goods*

Corollary

XOR-of-OR size $k \Rightarrow$ *OR* size* k, $\leq k^2$ *dummy goods*

Advanced topics in combinatorial auctions

- iterative combinatorial auction mechanisms
  - reduce the amount bidders have to disclose / communication complexity
  - allow bidders to learn about each others' valuations: e.g., affiliated values
- non-VCG mechanisms for restricted valuation classes
  - these can rely on polynomial-time winner determination algorithms