CS 6840 – Algorithmic Game Theory (6 pages)	Spring 2012
Lecture 31 Scribe Notes	
Instructor: Eva Tardos	*Aurosish Mishra (am2357)*

1 Recap: 2^{nd} Price Item Auction

Recall the setup for a 2^{nd} price item auction. We have a set of items, S, up for sale, and the following two assumptions about the bidders:

- Each player/bidder i has valuations for sets of items; $v_i(A) \geq 0$, for set A.
- The valuations are monotone or we have the so-called free-disposal assumption; $v_i(A) \leq v_i(B)$, if $A \subseteq B$

The auction format is such that players do not bid for sets, but for individual items, and the auctioneer runs a separate 2nd price auction for every item. Each player i submits bids $b_i(a)$ for an item a. The winner i is chosen such that $\max_i b_i(a) = p(a)$, which is defined as the price for this item. However, the winner i still pays the 2^{nd} price, i.e $\max_{j \neq i} b_j(a)$. The term $p(a)$ is just the price that someone else would have to pay to take over the item from the current winner. Given this setup, the optimum solution is given by: $\text{OPT} = \max \sum_i v_i(O_i)$ for a particular partition of items: $\{O_1, \ldots O_n\}$.

2 Proof Style: Previous 2 Lectures

We basically wanted to get a Price of Anarchy of 2 for this auction. Last class, we tried to prove this bound using a smoothness line of argument, and conclude this bound not just for pure Nash equilibriums, but also for learning algorithms. However, we did not finish the proof completely because of the fire drill.

Argument. We argued in the following manner. Let, $A_1, \ldots A_n$ be a pure Nash outcome. We are trying to institute a bid, if:

$$v_i(O_i) > \sum_{a \in O_i} p(a) \tag{1}$$

where, the prices are defined as above. *Any player i could bid slightly above the price on each item, and win the set O_i.* Using a Nash argument, we get the inequality that:

$$v_i(O_i) - \sum_{a \in O_i} p(a) \leq v_i(A_i) \tag{2}$$

where A_i is the set that player i wins at Nash equilibrium. Thus, we have a way of linking the optimum sets O_i with the Nash sets A_i. Note that, if it turns out that $v_i(O_i) < \sum_{a \in O_i} p(a)$, then the LHS of eqn. (2) is negative, and since the valuations of all sets are positive, the equation trivially holds (nothing of interest).

3 New Flavour of Analysis

We want to replace the first part of the previous argument (in italics). We would like to have a fixed set of bids s^* that gives us an optimal solution. We want to tell the player what to bid, independent of what everyone else is bidding. If we can do that, then this line of argument extends to learning algorithms; just make sure that the player does not regret bidding s^*. We still want to use a smoothness sort of a framework.

3.1 Special Case.

Now, consider the special case of **Fractionally Sub-Additive Valuations**. We call a valuation additive if: $v_i(A) = \sum_{a \in A} v_i(a)$. Now, for our special type of valuations, instead of each player having one such valuation, he has multiple ways of valuing a set (possibly based on the different kinds of use of that set). He chooses the valuation that is best for him. Mathematically, let us assume that player i has set vectors V_i, with

$$v_i(A) = \max_{v \in V_i} \sum_{a \in A} v(a)$$

This is a very general class of functions, which contains several other interesting function classes. For example, a function class with economy of scale (an additional item is of less value) is a less general class than this.

3.2 Modified Analysis: PoA

If v_i is fractionally sub-additive, we want to still be able to finish the proof in the smoothness framework. Note that we care only about the optimal sets. Using the definition of the valuation for this special class of functions, we have:

$$v_i(O_i) = \max_{v \in V_i} \sum_{a \in O_i} v(a) \tag{3}$$

Observe that any particular vector v is interesting, since each of them is exactly of the form we want to use for our bid.

Fact 1: *If each player bids $b_i = v$, where $v \in V_i$ is the vector maximizing the valuation for set O_i, we want to claim that the resulting allocation must be optimum.*

Proof. We want to show something along the lines of eqn. (2). We know for sure that:

$$v_i(O_i) \leq \sum_{a \in O_i} p(a) \tag{4}$$

This is easy to see as follows:

$$\begin{aligned} v_i(O_i) &= \sum_{a \in O_i} b_i(a) \\ &\leq \sum_{a \in O_i} p(a) \end{aligned}$$

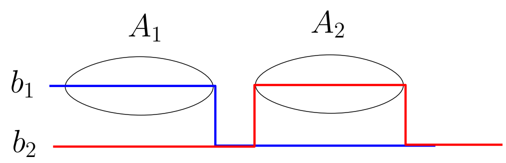

Figure 1: Possible Bids

The first equality follows from the way we chose our bids to be the valuation maximizing vector for the set O_i. The second inequality is true since $b_i(a)$ is just one of the bids, whereas $p(a)$ is the maximum bid on the item a. Summing over all bidders, we get:

$$\begin{aligned}\sum_i v_i(O_i) &\leq \sum_a p(a) \\ &= \sum_i \sum_{a \in A_i} b_i(a) \\ &\leq \sum_i v_i(A_i)\end{aligned}$$

The first inequality helps us eliminate the optimal sets from the prices. The second equality follows by thinking of the prices as the bids of players who win the item at Nash, assuming that $A_1, \ldots A_n$ is the allocation that the players get at Nash. The third inequality follows from the fact that for the set A_i that a player is winning, the b_i-vector might not be the determining vector, some other vector might determine the value of the set (Remember that the bid was just one of the vectors, See Fig. 1).

In our analysis, everything was pessimsitic; and we obtain the following inequality:

$$\sum_i v_i(O_i) \leq \sum_i v_i(A_i) \tag{5}$$

Since, we have chosen O_i to be the optimal allocation for player i, and any allocation A_i cannot be better than the optimum, we actually have equality. This means that the players were actually the highest for every single item, and the inequalities are actually equalities in the previous steps.

Comments. Thus, in the class of fractionally sub-additive valuation functions, there is a natural selection of what it means to get the optimum. Each player has a bid that doesn't depend on the bids of other players, rather it depends on other players only through the optimal partition of the items. The bid for any player is just his valuation maximizing vector, i.e. you are supposed to bid the linear vector that defines your value at the optimum.

3.2.1 Utility of Players

Suppose we have a Nash allocation: $A_1, A_2, \ldots, A_n$ using the bid vector $b = (b_1, b_2, \ldots, b_n)$. Let, ${b_i}^*$ be the optimum bid as defined in the previous section.

$$b_i{}^* = \text{bid from} \max_{v \in V_i} \sum_{a \in O_i} v(a)$$

$b_i{}^*(O_i) = v_i(O_i)$ and $b_i{}^* \in V_i$ (one of the linear vectors that has the right value at the optimum)

We want to think of the utility of bidding a deviation bid of $b_i{}^*$ at Nash: $u_i(b_i{}^*, b_{-i})$. The condition for Nash equilibrium says that the utility we get by deviating to an alternate bid should be no greater than the value of the allocation at Nash. Note that we may have to pay to get the Nash allocation, but still the utility will always be bounded by the value of the set. (Any price will be a positive number subtracted from the valuation. So, the inequality would still hold)

$$u_i(b_i{}^*, b_{-i}) \leq v_i(A_i) \tag{6}$$

As before, we want to say something like eqn. (2).

Claim:

$$\begin{aligned} u_i(b_i{}^*, b_{-i}) &\geq v_i(O_i) - \sum_{a \in O_i} p(a) \\ &= \sum_{a \in O_i} (b_i{}^*(a) - p(a)) \end{aligned}$$

By the choice of our bid vectors $b_i{}^*$, the value of the optimal set for player i can be written as the sum of bids on the items in the optimal set O_i. Thus, the claim can be restated as mentioned in the second line.

Simpler Argument. In order to prove the claim, first consider the following simpler argument regarding the utility of a player:

$$u_i(b_i{}^*, b_{-i}) \geq \sum_a (b_i{}^*(a) - p(a))^+$$

The + denotes that we consider only positive terms in the expression. Observe that the expression on the right represents exactly the items bidder i is going to win, since these are the items on which he is outbidding the current maximum bid (price). This seems like an equality, but it is correctly an inequality because in case you were the highest bidder, you might have gotten the item for a cheaper price. But you definitely get this much utility. This can be extended to say that:

$$u_i(b_i{}^*, b_{-i}) \geq \sum_{a \in O} (b_i{}^*(a) - p(a)) \tag{7}$$

This is true for any set O because adding negative terms would only make the RHS smaller, as would dropping some items (positive terms get dropped when we remove items).

3.2.2 Obtaining the Smoothness Framework

Summing over all players i, and using certain facts about our auction, we get from eqn. (7):

$$\sum_i u_i(b_i{}^*, b_{-i}) \geq \sum_i v_i(O_i) - \sum_i v_i(A_i) \tag{8}$$

This is easy to see as follows:

$$\begin{aligned}\sum_i u_i(b_i{}^*, b_{-i}) &\geq \sum_i v_i(O_i) - \sum_a p(a) \\ &= \sum_i v_i(O_i) - \sum_i \sum_{a \in A_i} b_i(a) \\ &\geq \sum_i v_i(O_i) - \sum_i v_i(A_i)\end{aligned}$$

The first inequality follows from eqn. (7), and the fact that the optimum sets are just a way to partition the entire set of items. The second equality follows from the simple fact that the guy who bid the maximum (defined by $p(a)$) is the winner of the item at Nash. The third inequality follows from the conservative assumptions on bidders, i.e. $\sum_{a \in A_i} b_i(a) \leq v_i(A_i)$.

Price of Anarchy Result. Summing eqn. (6) over all bidders, and combining it with eqn. (8), we get:

$$\sum_i v_i(A_i) \geq \sum_i v_i(O_i) - \sum_i v_i(A_i) \tag{9}$$

Rearranging the terms, we get a PoA bound of 2.

Comments. We have taken a pretty straightforward proof and fit it into a scheme in which Nash equilibrium has high quality if the players name a set of bids s^* that defines the optimum value. That is to say, at any point, if you bid by following this fixed bid s^*, you will win a high enough value. Assuming, we can pre-compute the OPT, we can tell each player what they should bid to get a high enough value.

4 Advantages of this Analysis

This analysis gives us two very straightforward advantages. First, it can be extended to a Mixed Nash kind of setting. Second, it will also work for learning algorithms.

4.1 Mixed Nash Setting

In a Mixed Nash equilibrium, players basically randomize between their several available options.

Old Analysis. Let us consider the old analysis framework, and see how it performs in a Mixed Nash setting. The bid is now random, which implies that the price is random. So, it is now not an option to bid slightly above the prices to beat it, since we don't know the price. We can however take expectation, and try to go over the expected price with our bid. We can then say things like with a certain probability a player can beat the price and with a certain other probability he can't do so. Consider a scenario in which we went twice above the expected price, and so have a 1/2 probability of bidding above the price. This gives us an extra factor of 4 in the analysis. So, the old line of analysis is not really suited for Mixed Nash settings.

New Analysis. In the new style of analysis, prices exist, but it won't matter because a player's bid doesn't depend on it. The Nash condition in eqn. (6) is no longer true in general, but it is true in expectation. This means that in any particular outcome, you might not be doing better than b_i^*, but in expectation you would do better. So, we have:

$$\mathbb{E}_{b_{-i}}[u_i({b_i}^*, b_{-i})] \leq \mathbb{E}_b[v_i(A_i)] \tag{10}$$

As a player, I can bid ${b_i}^*$ or b_i and watch the corresponding expectations, which are meaningful things to compute in this setting. In eqn. (10), we had to take expectation of the RHS as well since in the mixed Nash setting because bids are random, the allocations at Nash, A_i's, are also random. Observe that since we are at Nash, deterministically bidding b_i^* is not a good strategy.

The Claim leading to eqn. (8) is however, true for any particular set of prices, irrespective of what the bids were; it has got nothing to do with expectations. It is true for all incarnations of the random variables. However, now, the terms $\sum\limits_i u_i({b_i}^*, b_{-i})$ and $\sum\limits_i v_i(A_i)$ are both going to have random values. In order to get our PoA bound, we need to combine equations (10), and an expectation version of eqn. (8), which can be written as:

$$\mathbb{E}_b[\sum_i u_i({b_i}^*, b_{-i})] \geq \sum_i v_i(O_i) - \mathbb{E}_b[\sum_i v_i(A_i)] \tag{11}$$

The valuation of the optimum set for any player is a deterministic number, and hence stays the same after taking expectation. Observe that the eqn. (10) has the expectation term of the utility, taken over the bids of other players b_{-i}, but the eqn. (11) has the expectation term of the utility taken over the entire bid vector b. However, since the utility expressions for player i don't have any b_i terms, it is safe to restrict the expectation term in eqn. (11) to the bid vector b_{-i}; thereby, giving us the same PoA bound for the mixed Nash setting.

4.2 Learning Algorithms

This sort of analysis also works for learning algorithms. The eqn. (7) can be now interpreted as saying that, player i doesn't regret bidding ${b_i}^*$. In some sense, the special class of fractionally sub-additive valuations, tell us what strategies not to regret. All the linear vectors v are natural bid vectors. We don't know which vector we should choose, but as long as we do not regret it, we should be okay. In fact, we should learn which is the proper way to bid, by mixing between the possible bid vectors. If the set of vectors V_i is small enough, we can learn easily; if not, it gets slightly tricky.

5 Next Class

On Friday and Monday, we will cover the Generalized Second Price (GSP) auctions, which is used for selling advertisements on the internet, and is a widely prevalent application of our theoretic knowledge.