CS 6840: Algorithmic Game Theory

Spring 2017

Lecture 33: April 21

Lecturer: Éva Tardos Scribe: Seunghee Han

33.1 Recap

Recall from last lecture we had the following

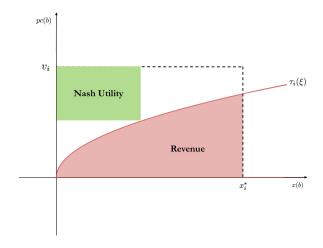


Figure 33.1: Price per click vs. Expected Clicks

and we proved the following bound on the empirical price of anarchy:

Theorem 33.1

$$u_i + \frac{1}{\mu} \int_0^{x_i^*} \tau_i(\xi) \ d\xi \ge \frac{1 - e^{-\mu}}{\mu} v_i x_i^*$$

This is useful as if we have $Rev \ge \frac{1}{\mu} \sum_i \int_0^{x_i^*} \tau_i(\xi) d\xi$, then we get a bound on the price of anarchy:

$$\sum_{i} u_{i} + Rev \ge \sum_{i} u_{i} + \frac{1}{\mu} \sum_{i} \int_{0}^{x_{i}^{*}} \tau_{i}(\xi) d\xi \ge \frac{1 - e^{-\mu}}{\mu} \sum_{i} v_{i} x_{i}^{*}$$

However, this theorem raises some questions

- How can we use this theorem when x_i^* is not in the data?
- Why are we comparing $\int_0^{x_i^*} \tau_i(\xi) d\xi$ and Revenue?

Today we will focus on answering these two questions.

33.2 Calculating Empricial Price of Anarchy

We want to show that

$$Rev \ge \frac{1}{\mu} \sum_{i} \int_{0}^{x_{i}^{*}} \tau_{i}(\xi) \ d\xi$$

but as previously mentioned, this is difficult since we do not know what x_i^* is.

Instead, we show

$$Rev \ge \frac{1}{\mu} \max_{x} \sum_{i} \int_{0}^{x} \tau_{i}(\xi) \ d\xi$$

where the x is ranging over all feasible vectors $x = (x_1, \dots, x_n)$. This is much easier to calculate and test.

This suffices to prove the empirical price of anarchy bound:

$$\sum_{i} u_i + Rev \ge \sum_{i} u_i + \frac{1}{\mu} \max_{x} \sum_{i} \int_{0}^{x_i} \tau_i(\xi) d\xi$$

$$\ge \sum_{i} u_i + \frac{1}{\mu} \sum_{i} \int_{0}^{x_i^*} \tau_i(\xi) d\xi$$

$$\ge \frac{1 - e^{-\mu}}{\mu} \sum_{i} v_i x_i^*$$

33.3 $(1-\frac{1}{e})$ Price of Anarchy Bound for First Price Auctions

Now let's consider this in a first price full info single item auction. This will also explain why comparing the integral to revenue makes sense.

In this case our function $\tau_i(\xi)$ is a monotone function like the blue function on the figure below.

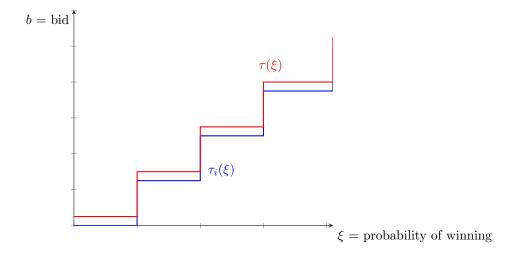


Figure 33.2: Bid vs. Probability of Winning for First Price Auctions

Lecture 33: April 21 33-3

Assume that players are numbered so that v_i is the highest value. Now we get that the optimal solution is $x^* = [1, 0, 0, ..., 0]$.

Therefore, we try to show

$$Rev \geq \frac{1}{\mu} \int_0^1 \tau_1(\xi) \ d\xi$$

Recall that $\tau_i(\xi)$ is the value i has to bid to get ξ probability of winning.

Let's define a function $\tau(\xi)$: the bid an extra person entering the auction needs to bid to win with probability ξ . For this extra person to win, she needs to also outbid all players currently in the auction. So we get that $\forall i \tau(\xi) \geq \tau_i(\xi)$, which is represented as red in Figure 33.2.

Claim 33.2 In first price auction:

$$\int_0^1 \tau(\xi) \ d\xi = Revenue$$

Proof: To make the argument more intuitive, assume $\tau(\xi)$ is a step-function, and consider the following graph:

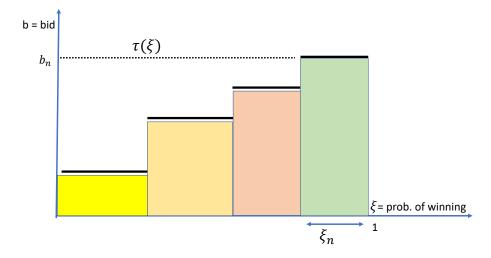


Figure 33.3: Bid vs. Probability of Winning

The green region is the probability that $\max_{i} b_{i} = b_{n}$, that is, bidding b_{n} guarantees that the extra person wins with probability 1. Since the maximum bid is the revenue in first price auctions, this region is the contribution of the revenue when the maximum bid is b_{n} .

The red region represents the probability that $\max_{i} b_i = b_{n-1}$. If the extra person bids b_{n-1} she is guaranteed to win with probability $1 - \xi_n$. Again, this region contributes to the revenue as well as the integral.

The same logic can be applied to the other regions as well.

Therefore, summing all the shaded regions is revenue. Thus, $\int_0^1 \tau(\xi) d\xi = Revenue$

Now, since $\forall i \ \tau(\xi) \geq \tau_i(\xi)$, $Revenue \geq \int_0^1 \tau_i(\xi) \ d\xi$ as we wanted.

Finally, using our theorem with $\mu = 1$, we have

$$\sum_{i} u_{i} + Rev = \sum_{i} u_{i} + \int_{0}^{1} \tau(\xi) d\xi \ge u_{i} + \int_{0}^{1} \tau(\xi) d\xi \ge (1 - \frac{1}{e}) OPT$$

proving a $(1 - \frac{1}{e})$ price of an archy bound for first price auctions.