CS599: Algorithm Design in Strategic Settings
Fall 2012
Lecture 11: Ironing and Approximate Mechanism Design in Single-Parameter Bayesian Settings

Instructor: Shaddin Dughmi

Outline

Outline

Recall: Single-item Allocation in Bayesian Setting

We considered Single-item Auctions.

Recall: Single-item Allocation in Bayesian Setting

We considered Single-item Auctions.

Bayesian Assumption

We assume each player's value is drawn independently from some distribution F_i.

Recall: Single-item Allocation in Bayesian Setting

We considered Single-item Auctions.

Bayesian Assumption

We assume each player's value is drawn independently from some distribution F_i.

We saught the BIC mechanism maximizing expected revenue.

Virtual Value

Definition

The virtual value $\phi(v)$ of a player with value v at quantile q is $R'(q)$, or equivalently:

$$\phi(v) = v - \frac{S(v)}{f(v)}$$

e.g. For U[0,1], $\phi(v) = 2v - 1$

Myerson's Revenue-Optimal Auction (Regular)

Lemma (Myerson's Virtual Surplus Lemma)

Let $M = (\mathcal{A}, p)$ be a BIC mechanism where a player bidding zero pays nothing in expectation. The expected revenue of M is equal to the expected virtual welfare served by $\mathcal{A}$.

Theorem

The revenue optimal BIC single-item auction awards the item to the player with the highest nonnegative virtual value, and discards the item if all virtual values are negative.

e.g. For i.i.d U[0,1], vickrey with a reserve price of $\phi^{-1}(0) = 0.5$.

Myerson's Revenue-Optimal Auction (Regular)

Lemma (Myerson's Virtual Surplus Lemma)

Let $M = (\mathcal{A}, p)$ be a BIC mechanism where a player bidding zero pays nothing in expectation. The expected revenue of M is equal to the expected virtual welfare served by $\mathcal{A}$.

Theorem

The revenue optimal BIC single-item auction awards the item to the player with the highest nonnegative virtual value, and discards the item if all virtual values are negative.

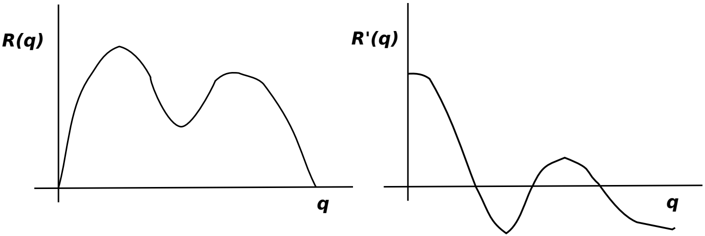
e.g. For i.i.d U[0,1], vickrey with a reserve price of $\phi^{-1}(0) = 0.5$.

In our proof, we assumed the distribution is regular.

- $\phi(v)$ monotone non-decreasing in v.
- Includes many natural distributions: uniform, normal, exponential. . .
- Does not include everything: bimodal, power-law . . .

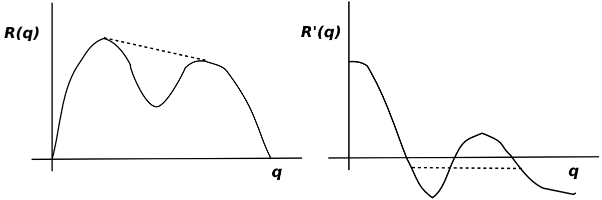
Outline

What Goes Wrong Without Regularity?



- Revenue non-concave $\iff$ virtual value non-monotone in value
- Choosing player with highest virtual value not necessarily monotone allocation rule

Ironing



Ironing

- The ironed revenue curve $\overline{R}$ is the concave closure of R.
- The ironed virtual value $\overline{\phi}$ is the derivative of $\overline{R}$.

Intuition

- To enforce monotonicity, "lump together" types in a non-concave region of R.
- Ironed VV averages virtual value in each group.
- Alternative interpretation: $\overline{R}(q)$ is the true maximum revenue possible if constrained to selling probability q.

Ironed VV vs VV

Lemma

In any monotone allocation rule, expected Ironed VV served $\geq$ expected VV served.

Because ironed revenue curve is point-wise higher than revenue curve at every offer price.

Ironed VV vs VV

Lemma

In any monotone allocation rule, expected Ironed VV served $\geq$ expected VV served.

Because ironed revenue curve is point-wise higher than revenue curve at every offer price.

Lemma

If a monotone allocation rule does not distinguish types in the same group, its expected virtual value served (i.e. revenue) is equal to its expected ironed virtual value served.

Ironed VV vs VV

Lemma

In any monotone allocation rule, expected Ironed VV served $\geq$ expected VV served.

Because ironed revenue curve is point-wise higher than revenue curve at every offer price.

Lemma

If a monotone allocation rule does not distinguish types in the same group, its expected virtual value served (i.e. revenue) is equal to its expected ironed virtual value served.

Theorem

The allocation rule maximizing ironed virtual value is montone, and maximizes expected revenue. (when combined with Myerson payments)

Myerson's Revenue-Optimel Auction (General)

Theorem

The revenue optimal BIC single-item auction awards the item to the player with the highest nonnegative ironed virtual value, breaking ties independently of value, and discards the item if all ironed virtual values are negative.

Outline

1. Recap
2. Non-regular Distributions: Ironing
3. Implications for Single-Parameter Problems
4. A Reduction from Approximation Algorithms to Truthful Mechanisms

Revenue-Optimal Mechanisms for Single-Parameter Problems

Our proof didn't use any structure particular to the single item auction.

Theorem

For any single-parameter problem, where player's private parameters are drawn independently, the revenue-maximizing auction is that which maximizes ironed virtual welfare. Specifically, with the allocation rule

$$\mathcal{A}(v) = \operatorname*{argmax}_{x \in \Omega} \sum_i \overline{\phi}_i(v_i) x_i$$

Revenue-Optimal Mechanisms for Single-Parameter Problems

Our proof didn't use any structure particular to the single item auction.

Theorem

For any single-parameter problem, where player's private parameters are drawn independently, the revenue-maximizing auction is that which maximizes ironed virtual welfare. Specifically, with the allocation rule

$$\mathcal{A}(v) = \underset{x \in \Omega}{\operatorname{argmax}} \sum_i \overline{\phi}_i(v_i) x_i$$

Examples

- k-item Auction
- Position Auctions
- Matching (binary service, weighted separable)

Approximately Revenue-Optimal Mechanisms

We have identified the revenue optimal mechanism for arbitrary single-parameter problems, however this is not helpful for problems where [virtual] welfare maximization is NP-hard

- e.g. Single-minded CA, Knapsack

Approximately Revenue-Optimal Mechanisms

We have identified the revenue optimal mechanism for arbitrary single-parameter problems, however this is not helpful for problems where [virtual] welfare maximization is NP-hard

- e.g. Single-minded CA, Knapsack

Corollary

If a single parameter problem admits a polynomial time DSIC α*-approximation (worst case) mechanism for welfare, then it also admits a polynomial-time DSIC* α*-approximation (average case) mechanism for revenue.*

- e.g. we saw $\sqrt{m}$ for Single-minded CA, 2 for Knapsack

Outline

BIC Approximate Mechanisms for Single-Parameter Problems

So far, when approximation was necessary, we have designed IC mechanisms carefully catered to the problem. It is unknown how to get around that for DSIC.

Does relaxing the the Bayesian Setting, and requiring only BIC, buy us any more?

BIC Approximate Mechanisms for Single-Parameter Problems

So far, when approximation was necessary, we have designed IC mechanisms carefully catered to the problem. It is unknown how to get around that for DSIC.

Does relaxing the the Bayesian Setting, and requiring only BIC, buy us any more?

Theorem (Hartline, Lucier 10)

For any single-parameter problem where player values are drawn independently from a product distribution F supported on $[0, h]^n$, any allocation algorithm $\mathcal{A}$, any parameter ϵ, there is a BIC algorithm $\overline{\mathcal{A}}_\epsilon$ that preserves the average case welfare of $\mathcal{A}$ up to an additive ϵ, and moreover can be implemented in time polynomial in $n, \log h$, and $\frac{1}{\epsilon}$.

Will ignore sampling issues and get rid of ϵ, just to get the idea.

Recall: Myerson's Monotonicity Lemma

Let x_i and p_i be a function of the quantile of the player's report rather than the report itself.

Myerson's Monotonicity Lemma (BIC)

Consider a mechanism for a single-parameter problem in a Bayesian setting where player values are independent. Let $x_i(q_i)$ and $p_i(q_i)$ be the interim allocation/payment rules faced by player i when other players play the truth-telling strategy. The mechanism is BIC if and only if:

- $x_i(q_i)$ is a monotone non-increasing function of q_i
- $p_i(q_i)$ is an integral of $v_i(q_i)dx_i = v_i(q_i)x_i'(q_i)dq_i$. Doing the integration:

$$p_i(q_i) = \int_{r=q_i}^{1} v_i(r)x_i'(r)dr$$

Ironing A Single-Player Interim Allocation Rule

Non-BIC algorithm $\mathcal{A}$

Interim rule $x_i(q_i)$ not monotone decreasing as needed.

Ironing A Single-Player Interim Allocation Rule

Non-BIC algorithm $\mathcal{A}$

Interim rule $x_i(q_i)$ not monotone decreasing as needed.

Ironing Allocation Rule x

- Let $X(q) = \int_{q'=0}^{q} x(q)$ be the cumulative allocation rule.
  - Expected quantity player bidding above $v(q)$ gets.
  - Concave iff x monotone decreasing.
- Let $\overline{X}$ be the concave closure of X. Always above X.
- The ironed allocation rule $\overline{x}(q) = \frac{dX}{dq}(q)$ is the derivative of $\overline{X}$.

Ironing A Single-Player Interim Allocation Rule

Non-BIC algorithm $\mathcal{A}$

Interim rule $x_i(q_i)$ not monotone decreasing as needed.

Ironing Allocation Rule x

- Let $X(q) = \int_{q'=0}^{q} x(q)$ be the cumulative allocation rule.
  - Expected quantity player bidding above $v(q)$ gets.
  - Concave iff x monotone decreasing.
- Let $\overline{X}$ be the concave closure of X. Always above X.
- The ironed allocation rule $\overline{x}(q) = \frac{dX}{dq}(q)$ is the derivative of $\overline{X}$.

Implement

Allocation rule of the ironed algorithm $\overline{\mathcal{A}}$ simply replaces any bid in an ironed interval with a random bid drawn from that interval (from that player's distribution), before calling $\mathcal{A}$.

Ignoring: sampling issues identifying ironing intervals

Ironing Allocation Rule increases welfare

Fact

If a function $\overline{x}$ is such that its cumulative integral exceeds that of x at every point, then for any decreasing function w we have $\int_q \overline{x}(q)w(q) \geq \int_q x(q)w(q)$

The weighting $w(q) = v(q)$ is decreasing, so expected welfare increases <u>on average</u>.

Are we done?

Wrinkle

We showed how to iron a single player's allocation rule. Need to do all simultaneously...

But re-drawing a player's type before calling $\mathcal{A}$ changes other player's interim allocation rule...

Are we done?

Wrinkle

We showed how to iron a single player's allocation rule. Need to do all simultaneously...
But re-drawing a player's type before calling $\mathcal{A}$ changes other player's interim allocation rule...

Question

How can we iron all players' interim allocation rules simultaneously, preserving monotonicity?

Are we done?

Wrinkle

We showed how to iron a single player's allocation rule. Need to do all simultaneously...
But re-drawing a player's type before calling $\mathcal{A}$ changes other player's interim allocation rule...

Question

How can we iron all players' interim allocation rules simultaneously, preserving monotonicity?

Answer

We already did! From each player i's perspective, distribution of j's bids plugged into the algorithm unchanged!

Wrapup

Combined with the usual payment computation tricks, plus tricks to handle sampling issues, yields

Theorem (Hartline, Lucier 10)

For any single-parameter problem where player values are drawn independently from a product distribution F supported on $[0, h]^n$, any allocation algorithm $\mathcal{A}$, any parameter ϵ, there is a BIC algorithm $\overline{\mathcal{A}}_\epsilon$ that preserves the average case welfare of $\mathcal{A}$ up to an additive ϵ, and moreover can be implemented in time polynomial in n, $\log h$, and $\frac{1}{\epsilon}$.

Therefore, ignoring the additive loss of ϵ,

- A worst-case α-approximation algorithm for welfare implies an average case α-approximation mechanism for welfare or revenue.
- An average case α-approximation algorithm for welfare implies an average case α-approximation mechanism for welfare (not revenue)