Mechanisms & Auctions

Lawrence & Willi

Algorithmic Game Theory, 2017

Outline

Mechanism Design

Mechanism Design

Definition

A **Bayesian game setting** is a tuple (N, O, Θ, p, u).

- N is a set of n agents.
- $O = X \times \mathbb{R}^n$ is a set of outcomes (choice + payments).
- $\Theta = \Theta_1 \times \Theta_2 \times \cdots \times \Theta_n$ is a set of possible type vectors.
- p is a probability distribution on Θ.
- $u = (u_1, ..., u_n)$, $u_i : O \times \Theta \mapsto \mathbb{R}$ is the utility for agent i.

Mechanism Design

Definition

A **(quasilinear) mechanism** for $(N, O = X \times \mathbb{R}^n, \Theta, p, u)$ is a triple (A, χ, ρ).

- $A = A_1 \times \cdots \times A_n$, where A_i is the set of actions for agent i.
- $\chi : A \mapsto \Pi(X)$ maps each action profile to a distribution over choices.
- $\rho : A \mapsto \mathbb{R}^n$ maps each action profile to a payment for each agent.

Auctions in General

Auction Definition

Auction

- item A to be sold
- seller s who wants to sell A
- n bidders who want to buy A
- valuation v_i of getting A
- bid b_i for A
- price p_i for each bidder
- utility $u_i = v_i - p_i$

Auction Definition

Auction

- item A to be sold
- seller s who wants to sell A
- n bidders who want to buy A
- valuation v_i of getting A
- bid b_i for A
- price p_i for each bidder
- utility $u_i = v_i - p_i$

Mechanism

- $x = (0, ..., 0, 1, 0, ..., 0)$

Auction Definition

Auction

- item A to be sold
- seller s who wants to sell A
- n bidders who want to buy A
- valuation v_i of getting A
- bid b_i for A
- price p_i for each bidder
- utility $u_i = v_i - p_i$

Mechanism

- $x = (0, ..., 0, 1, 0, ..., 0)$
- agent i_0 (extra agent)

Auction Definition

Auction

- item A to be sold
- seller s who wants to sell A
- n bidders who want to buy A
- valuation v_i of getting A
- bid b_i for A
- price p_i for each bidder
- utility $u_i = v_i - p_i$

Mechanism

- $x = (0, ..., 0, 1, 0, ..., 0)$
- agent i_0 (extra agent)
- agents $i_1, ..., i_n$

Auction Definition

Auction

- item A to be sold
- seller s who wants to sell A
- n bidders who want to buy A
- valuation v_i of getting A
- bid b_i for A
- price p_i for each bidder
- utility $u_i = v_i - p_i$

Mechanism

- $x = (0, ..., 0, 1, 0, ..., 0)$
- agent i_0 (extra agent)
- agents $i_1, ..., i_n$
- Θ_i (private information)

Auction Definition

Auction

- item A to be sold
- seller s who wants to sell A
- n bidders who want to buy A
- valuation v_i of getting A
- bid b_i for A
- price p_i for each bidder
- utility $u_i = v_i - p_i$

Mechanism

- $x = (0, ..., 0, 1, 0, ..., 0)$
- agent i_0 (extra agent)
- agents $i_1, ..., i_n$
- Θ_i (private information)
- action $a_i \in A_i$

Auction Definition

Auction

- item A to be sold
- seller s who wants to sell A
- n bidders who want to buy A
- valuation v_i of getting A
- bid b_i for A
- price p_i for each bidder
- utility $u_i = v_i - p_i$

Mechanism

- $x = (0, ..., 0, 1, 0, ..., 0)$
- agent i_0 (extra agent)
- agents $i_1, ..., i_n$
- Θ_i (private information)
- action $a_i \in A_i$
- $\rho_i \in \rho$
- $u_i = u_i(\rho_i, \Theta_i)$

English Auction

English Auction

price $p \leftarrow$ low price from seller s
while item A not sold **do**
    $p \leftarrow$ bidder i's bid ($> p$)
    for all bidder j **do**
        if $p > v_j$ **then**
            bidder j drops out
        end if
    end for
    if only bidder i left **then**
        return i gets A for price p
    end if
end while

Dutch Auction

```
price p ← high price from seller s
while item A not sold do
    for all bidder i do
        if p ≤ v_i then
            bidder i bids p
            return i gets A for price p
        end if
    end for
    p ← p′ (p′ < p)
end while
```

Chinese Auction

Chinese Auction

for all bidder i **do**

  bidder i buys v_i tickets

end for

winner $\leftarrow$ seller s draws one ticket with name of bidder j

return winner gets A

Achieving Social Optimum

Achieving Social Optimum

- strategically complex mechanisms so far
- selfish behavior
- private information
- $\Rightarrow$ How to achieve social welfare? (Adam Smith's "invisible hand")
- $x^{opt}(v) = \text{argmax}_{x \in X} \sum_{i=1}^{n} v_i(x)$

Vickrey Auctions

Vickrey Auction

Motivation

How to assign good to player with highest valuation, not best strategy?

Vickrey Auction

Motivation

How to assign good to player with highest valuation, not best strategy?

Example: The government wants to auction off a wireless spectrum. The goal is **not to maximize their profit** but to get it in the hands of companies that **value it the most**, providing the best technology to the customers.

Vickrey Auction

Vickrey Auction

- bidders submit simultaneous sealed bids with b to the seller
- seller s opens them
- highest bidder gets item
- highest bidder pays second highest bid

Vickrey Auction Properties

Theorem

Bidding your true value v_i in a Vickrey auction is a dominant strategy.

Proof

Assume bidder i is bidding $b_i = v_i$.

In which situation would i receive a higher payoff by deviating from this strategy?

Vickrey Auction Properties

Theorem

Bidding your true value v_i in a Vickrey auction is a dominant strategy.

Proof

Assume bidder i is bidding $b_i = v_i$.

In which situation would i receive a higher payoff by deviating from this strategy?

	increase b_i	decrease b_i
already won item	**No**	**No** (doesn't change payoff; decreases chance to win)
already lost item	**No** (negative payoff)	**No**

Revenue Equivalence

Revenue Equivalence

1st price sealed bid	Dutch
Vickrey	English

Revenue Equivalence

Revenue Equivalence

"pays own bid"	1st price sealed bid	Dutch
"pays bid of 2^{nd}"	Vickrey	English

Revenue Equivalence

Revenue Equivalence

	strategically equivalent	
"pays own bid"	1st price sealed bid	Dutch
"pays bid of 2^{nd}"	Vickrey	English

Revenue Equivalence

Revenue Equivalence

strategically equivalent

"pays own bid"	1st price sealed bid	Dutch
"pays bid of 2nd"	Vickrey	English

revenue equivalent

Revenue Equivalence Theorem

Theorem

Any efficient auction mechanism yields the same expected revenue.

Setting

- efficient: the mechanism selects a choice x such that $\forall v \forall x', \sum_i v_i(x) \geq \sum_i v_i(x')$
- n risk-neutral players
- privately known value drawn independently from a common, strictly increasing distribution
- any bidder with lowest possible value expects zero utility

Revenue Equivalence Theorem - Outcome

Outcome

- i - bidder
- v_i - i's true value drawn from $[\underline{v}, \bar{v}]$
- $u_i(v_i)$ - expected utility of true value v_i
- $P_i(v_i)$ - probability of being awarded the good
- $\Rightarrow u_i(v_i) = u_i(\underline{v}) + \int_{x=\underline{v}}^{v_i} P_i(x)dx$

Revenue Equivalence Theorem - Outcome

Outcome

- i - bidder
- v_i - i's true value drawn from $[\underline{v}, \bar{v}]$
- $u_i(v_i)$ - expected utility of true value v_i
- $P_i(v_i)$ - probability of being awarded the good
- $\Rightarrow u_i(v_i) = u_i(\underline{v}) + \int_{x=\underline{v}}^{v_i} P_i(x)dx$
- mechanisms are efficient $\Rightarrow P_i$ always the same
- $\Rightarrow$ ***i*'s expected payment must be the same; independent of mechanism**

VCG Mechanism

Efficient Mechanisms

Definition

A quasilinear mechanism is **efficient**, if in equilibrium it selects a choice x such that

$$\forall v \forall x' : \sum_i v_i(x) \geq \sum_i v_i(x')$$

That is, an efficient mechanism selects the choice that maximizes the sum of agents utilities, disregarding the monetary payments that agents are required to make.

Groves Mechanism

Groves Mechanism

Groves mechanisms are direct efficient quasilinear mechanisms (χ, ρ).

$$\chi(\hat{v}) = \text{argmax}_x \sum_i \hat{v}_i(x),$$

$$\rho(\hat{v}) = h_i(\hat{v}_{-i}) - \sum_{j \neq i} \hat{v}_j(\chi(\hat{v}))$$

Groves Mechanism

Groves Mechanism

Groves mechanisms are direct efficient quasilinear mechanisms (χ, ρ).

$$\chi(\hat{v}) = \mathrm{argmax}_x \sum_i \hat{v}_i(x),$$
$$\rho(\hat{v}) = h_i(\hat{v}_{-i}) - \sum_{j \neq i} \hat{v}_j(\chi(\hat{v}))$$

$\Rightarrow \chi(\hat{v})$ is the social choice

Groves Mechanism

Groves Mechanism

Groves mechanisms are direct efficient quasilinear mechanisms (χ, ρ).

$$\chi(\hat{v}) = \text{argmax}_x \sum_i \hat{v}_i(x),$$

$$\rho(\hat{v}) = h_i(\hat{v}_{-i}) - \sum_{j \neq i} \hat{v}_j(\chi(\hat{v}))$$

$\Rightarrow \chi(\hat{v})$ is the social choice

$\Rightarrow \rho(\hat{v})$ is *"what would happen without i* - *what would happen with i"*

Clarke Tax

Clarke Tax

The **Clarke tax** sets the h_i term in a Groves mechanism as

$$h_i(\hat{v}_{-i}) = \sum_{j \neq i} \hat{v}_j(\chi(\hat{v}_{-i}))$$

Clarke Tax

Clarke Tax

The **Clarke tax** sets the h_i term in a Groves mechanism as

$$h_i(\hat{v}_{-i}) = \sum_{j \neq i} \hat{v}_j(\chi(\hat{v}_{-i}))$$

$\Rightarrow$ sum of all players' valuations (except i) of the optimal solution without i

VCG Mechanism

VCG Mechanism

The **Vickrey-Clarke-Groves Mechanism** is a direct efficient quasilinear mechanism (χ, ρ).

$$\chi(\hat{v}) = \text{argmax}_x \sum_i \hat{v}_i(x),$$

$$\rho(\hat{v}) = \sum_{j \neq i} \hat{v}_j(\chi(\hat{v}_{-i})) - \sum_{j \neq i} \hat{v}_j(\chi(\hat{v}))$$

VCG Mechanism

VCG Mechanism

The **Vickrey-Clarke-Groves Mechanism** is a direct efficient quasilinear mechanism (χ, ρ).

$$\chi(\hat{v}) = \mathrm{argmax}_x \sum_i \hat{v}_i(x),$$

$$\rho(\hat{v}) = \sum_{j \neq i} \hat{v}_j(\chi(\hat{v}_{-i})) - \sum_{j \neq i} \hat{v}_j(\chi(\hat{v}))$$

$\Rightarrow$ sum of all players' valuations (except i) of the optimal solution without i - sum of all players' valuations (except i) of the overall optimal solution

VCG Mechanism

VCG Mechanism

The **Vickrey-Clarke-Groves Mechanism** is a direct efficient quasilinear mechanism (χ, ρ).

$$\chi(\hat{v}) = \text{argmax}_x \sum_i \hat{v}_i(x),$$

$$\rho(\hat{v}) = \sum_{j \neq i} \hat{v}_j(\chi(\hat{v}_{-i})) - \sum_{j \neq i} \hat{v}_j(\chi(\hat{v}))$$

$\Rightarrow$ sum of all players' valuations (except i) of the optimal solution without i - sum of all players' valuations (except i) of the overall optimal solution

VCG Mechanism

VCG Mechanism

The **Vickrey-Clarke-Groves Mechanism** is a direct efficient quasilinear mechanism (χ, ρ).

$$\chi(\hat{v}) = \text{argmax}_x \sum_i \hat{v}_i(x),$$

$$\rho(\hat{v}) = \sum_{j \neq i} \hat{v}_j(\chi(\hat{v}_{-i})) - \sum_{j \neq i} \hat{v}_j(\chi(\hat{v}))$$

$\Rightarrow$ sum of all players' valuations (except i) of the optimal solution without i - sum of all players' valuations (except i) of the overall optimal solution $\Rightarrow$ **Social Cost**

Seller

Some company with ad slots for sale

Ad slot S1

500 clicks per week

Ad slot S2

300 clicks per week

Ad slot S3

100 clicks per week

Bidder A1

valuation $v_1 = 5$

Bidder A2

valuation $v_2 = 4$

Bidder A3

valuation $v_3 = 3$

Bidder A4

valuation $v_4 = 2$

Bidder A5

valuation $v_5 = 1$

VCG Truthfulness

Theorem

Truth telling is a dominant strategy under any Groves Mechanism.

VCG Truthfulness

Theorem

Truth telling is a dominant strategy under any Groves Mechanism.

Groves Mechanism

Groves mechanisms are direct efficient quasilinear mechanisms (χ, ρ).

$$\chi(\hat{v}) = \mathrm{argmax}_x \sum_i \hat{v}_i(x),$$

$$\rho(\hat{v}) = h_i(\hat{v}_{-i}) - \sum_{j \neq i} \hat{v}_j(\chi(\hat{v}))$$

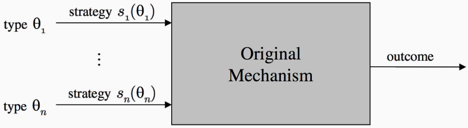
$\Rightarrow \chi(\hat{v})$ is the social choice

$\Rightarrow \rho(\hat{v})$ is *"what would happen without i - what would happen with i"*

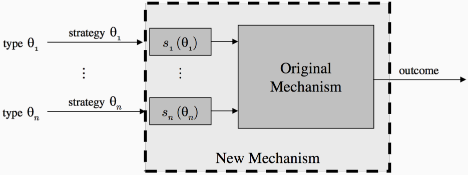
Revelation Principle

Theorem

If there exists any mechanism that implements a social choice function C in dominant strategies then there exists a direct mechanism that implements C in dominant strategies and is truthful.



(a) Revelation principle: original mechanism



(b) Revelation principle: new mechanism

Revelation Principle

Proof

Original mechanism:

dominant strategies s_1 to s_n

New mechanism:

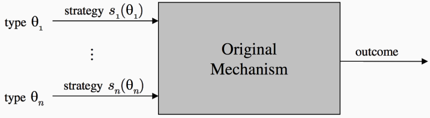
utility functions u_i

determine s_1 to s_n

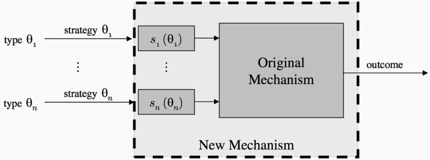
choose outcome from s_1 to s_n

Contradiction:

If i were better of lying with u_i* instead of u_i, he would prefer to follow s_i* in original mechanism rather than s_i ↯ as s_i is dominant.



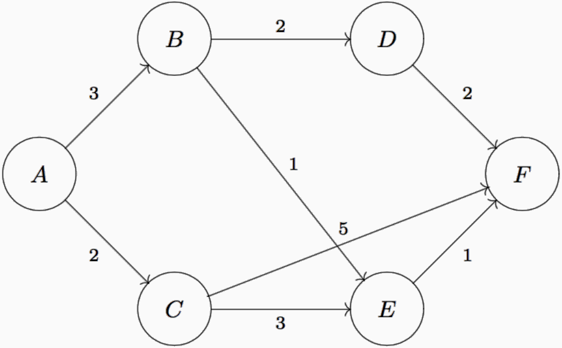
(a) Revelation principle: original mechanism



(b) Revelation principle: new mechanism

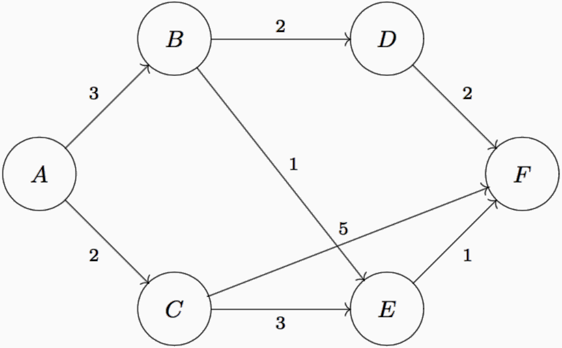
VCG for Shortest Paths

Send message from A to F



VCG for Shortest Paths

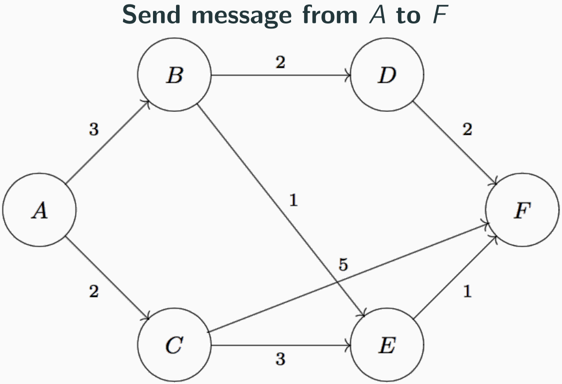
Send message from A to F



Setting:

- each edge e is an agent
- edge weight is $cost_e$ for agent e
- $u_i = -\text{cost}_e$
- mechanism pays agents $\rightarrow -p$

VCG for Shortest Paths



Setting:

- each edge e is an agent
- edge weight is $cost_e$ for agent e
- $u_i = -\text{cost}_e$
- mechanism pays agents $\rightarrow -p$

Payments:

$\rho_{AB} = (-6) - (-2) = -4$

$\rho_{BE} = (-6) - (-4) = -2$

$\rho_{EF} = (-7) - (-4) = -3$

VCG Downsides

Disclosure

Agents have to disclose private information.

VCG Downsides

Disclosure

Agents have to disclose private information.

Not Frugal

Does not choose the cheapest solution.

VCG Downsides

Disclosure

Agents have to disclose private information.

Not Frugal

Does not choose the cheapest solution.

Collusion

Susceptible to group collusion (truthful not dominant strategy).

VCG is Collusion Susceptible

Agent	U(build road)	U(do not build road)	Payment
1	200	0	150
2	100	0	50
3	0	250	0

Figure 1: Building a road

VCG is Collusion Susceptible

Agent	U(build road)	U(do not build road)	Payment
1	200	0	150
2	100	0	50
3	0	250	0

Figure 1: Building a road

Agent	U(build road)	U(do not build road)	Payment
1	250	0	100
2	150	0	0
3	0	250	0

Figure 2: Colluded road building

VCG Downsides

Disclosure

Agents have to disclose private information.

Not Frugal

Does not choose the cheapest solution.

Collusion

Susceptible to group collusion (truthful not dominant strategy).

Bidder Dropping

Dropping bidders can increase revenue.

VCG's Revenue Decreses with Bidder Dropping

Agent	U(build road)	U(do not build road)	Payment
1	0	90	0
2	100	0	90

Figure 3: Building a road

VCG's Revenue Decreses with Bidder Dropping

Agent	U(build road)	U(do not build road)	Payment
1	0	90	0
2	100	0	90

Figure 3: Building a road

Agent	U(build road)	U(do not build road)	Payment
1	0	90	0
2	100	0	0
3	100	0	0

Figure 4: Dropping players

Optimal Auctions for Bidders/Sellers

Optimal Auctions

- Sellers: Revenue Equivalence Theorem shows that differences between auctions vanish
- Bidders:

Optimal Auctions for Bidders/Sellers

Optimal Auctions

- Sellers: Revenue Equivalence Theorem shows that differences between auctions vanish
- Bidders: Difficult to determine

Dollar-Auction-Game

Dollar-Auction-Game

Best Strategy?

- 1 Dollar is auctioned off
- highest bid wins the dollar and pays his bid
- second bid doesn't win dollar but also pays his bid
- bid in multiples of 5 cents

Dollar-Auction-Game

Best Strategy?

- 1 Dollar is auctioned off
- highest bid wins the dollar and pays his bid
- second bid doesn't win dollar but also pays his bid
- bid in multiples of 5 cents
- $\Rightarrow$ don't play the game (if there are more players than yourself)