Algorithmic game theory

Martin Balko

11th lecture

December 21st 2023

Myerson's lemma

What we learned last time

What we learned last time

- For single-item auctions we have found an awesome auction (Vickrey's auction) with the following three properties:

What we learned last time

- For single-item auctions we have found an awesome auction (Vickrey's auction) with the following three properties:
  - DSIC: everybody has dominant strategy "bid truthfully" which guarantees non-negative utility,

What we learned last time

- For single-item auctions we have found an awesome auction (Vickrey's auction) with the following three properties:
  - DSIC: everybody has dominant strategy "bid truthfully" which guarantees non-negative utility,
  - strong performance: maximizing social surplus,

What we learned last time

- For single-item auctions we have found an awesome auction (Vickrey's auction) with the following three properties:
  - DSIC: everybody has dominant strategy "bid truthfully" which guarantees non-negative utility,
  - strong performance: maximizing social surplus,
  - computational efficiency: running in polynomial time.

What we learned last time

- For single-item auctions we have found an awesome auction (Vickrey's auction) with the following three properties:
  - DSIC: everybody has dominant strategy "bid truthfully" which guarantees non-negative utility,
  - strong performance: maximizing social surplus,
  - computational efficiency: running in polynomial time.
- We then generalized single-item auctions to single-parameter environments.

What we learned last time

- For single-item auctions we have found an awesome auction (Vickrey's auction) with the following three properties:
  - DSIC: everybody has dominant strategy "bid truthfully" which guarantees non-negative utility,
  - strong performance: maximizing social surplus,
  - computational efficiency: running in polynomial time.
- We then generalized single-item auctions to single-parameter environments.
  - 1 seller and n bidders, the seller collects the bids $b = (b_1, \ldots b_n)$,

What we learned last time

- For single-item auctions we have found an awesome auction (Vickrey's auction) with the following three properties:
  - DSIC: everybody has dominant strategy "bid truthfully" which guarantees non-negative utility,
  - strong performance: maximizing social surplus,
  - computational efficiency: running in polynomial time.
- We then generalized single-item auctions to single-parameter environments.
  - 1 seller and n bidders, the seller collects the bids $b = (b_1, \ldots b_n)$,
  - the seller chooses an allocation $x(b) = (x_1(b), \ldots, x_n(b))$ from X,

What we learned last time

- For single-item auctions we have found an awesome auction (Vickrey's auction) with the following three properties:
  - DSIC: everybody has dominant strategy "bid truthfully" which guarantees non-negative utility,
  - strong performance: maximizing social surplus,
  - computational efficiency: running in polynomial time.
- We then generalized single-item auctions to single-parameter environments.
  - 1 seller and n bidders, the seller collects the bids $b = (b_1, \ldots b_n)$,
  - the seller chooses an allocation $x(b) = (x_1(b), \ldots, x_n(b))$ from X,
  - the seller sets payments $p(b) = (p_1(b), \ldots, p_n(b))$.

What we learned last time

- For single-item auctions we have found an awesome auction (Vickrey's auction) with the following three properties:
  - DSIC: everybody has dominant strategy "bid truthfully" which guarantees non-negative utility,
  - strong performance: maximizing social surplus,
  - computational efficiency: running in polynomial time.
- We then generalized single-item auctions to single-parameter environments.
  - 1 seller and n bidders, the seller collects the bids $b = (b_1, \ldots b_n)$,
  - the seller chooses an allocation $x(b) = (x_1(b), \ldots, x_n(b))$ from X,
  - the seller sets payments $p(b) = (p_1(b), \ldots, p_n(b))$.
- Is there awesome mechanism (x, p) for single-parameter environments?

What we learned last time

- For single-item auctions we have found an awesome auction (Vickrey's auction) with the following three properties:
  - DSIC: everybody has dominant strategy "bid truthfully" which guarantees non-negative utility,
  - strong performance: maximizing social surplus,
  - computational efficiency: running in polynomial time.
- We then generalized single-item auctions to single-parameter environments.
  - 1 seller and n bidders, the seller collects the bids $b = (b_1, \dots b_n)$,
  - the seller chooses an allocation $x(b) = (x_1(b), \dots, x_n(b))$ from X,
  - the seller sets payments $p(b) = (p_1(b), \dots, p_n(b))$.
- Is there awesome mechanism (x, p) for single-parameter environments?
- We started by looking for DSIC mechanisms.

Myerson's lemma

Myerson's lemma

- A powerful tool for designing DSIC mechanisms.

Myerson's lemma

- A powerful tool for designing DSIC mechanisms.

Figure: Roger Myerson (born 1951) receiving a Nobel prize in economics.

Sources: https://en.wikipedia.org and https://twitter.com

Statement of Myerson's lemma

Statement of Myerson's lemma

- An allocation rule x is implementable if there is a payment rule p such that (x, p) is DSIC.

Statement of Myerson's lemma

- An allocation rule x is implementable if there is a payment rule p such that (x, p) is DSIC.
- An allocation rule x is monotone if, for every bidder i and all bids b_{-i}, the allocation $x_i(z; b_{-i})$ is nondecreasing in z.

Statement of Myerson's lemma

- An allocation rule x is implementable if there is a payment rule p such that (x, p) is DSIC.
- An allocation rule x is monotone if, for every bidder i and all bids b_{-i}, the allocation $x_i(z; b_{-i})$ is nondecreasing in z.

Myerson's lemma

In a single-parameter environment, the following three claims hold.

(a) An allocation rule is implementable if and only if it is monotone.

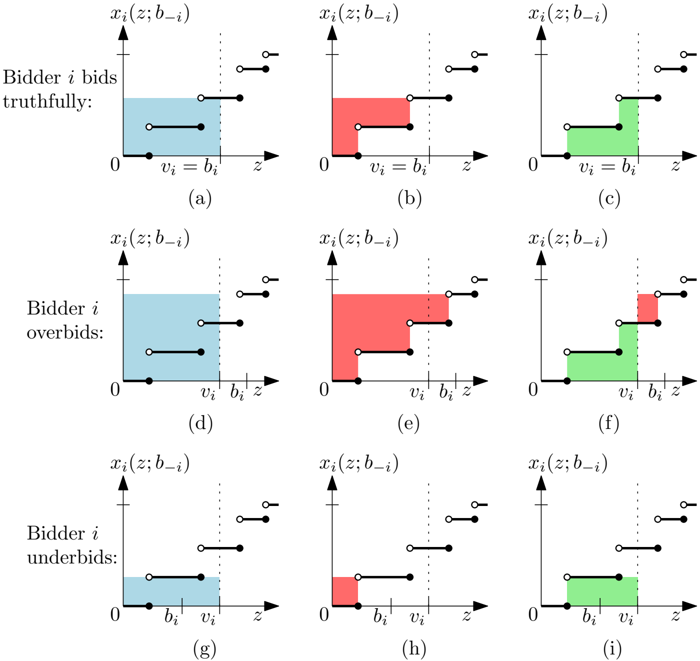
(b) If an allocation rule x is monotone, then there exists a unique payment rule p such that (x, p) is DSIC (assuming $b_i = 0$ implies $p_i(b) = 0$).

(c) For every i, the payment rule p is given by the following explicit formula

$$p_i(b_i; b_{-i}) = \int_0^{b_i} z \cdot \frac{\mathrm{d}}{\mathrm{d}z} x_i(z; b_{-i}) \, \mathrm{d}z.$$

Proof of Myerson's lemma

Proof of Myerson's lemma



Auction example: scheduling TV commercials

Auction example: scheduling TV commercials

- Bidders are companies such that each company has its own TV commercial of length w_i and is willing to pay v_i in order to have the commercial presented during a commercial break. The seller is a television station with a commercial break of length W.

Sources: https://mountain.com/ and https://www.eq-international.com/

Auction example: scheduling TV commercials

- Bidders are companies such that each company has its own TV commercial of length w_i and is willing to pay v_i in order to have the commercial presented during a commercial break. The seller is a television station with a commercial break of length W.

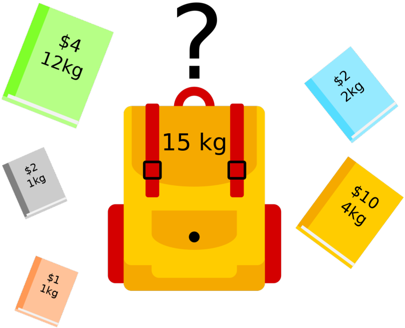
Sources: https://mountain.com/ and https://www.eq-international.com/

- Can we design an awesome mechanism that assigns the slots?

Knapsack problem

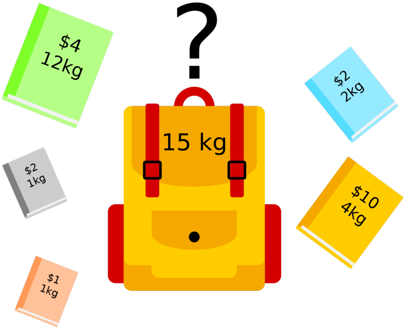
Knapsack problem

- given a capacity W and n items of values $v_1, \ldots, v_n$ and sizes $w_1, \ldots, w_n$, find a subset of the items having a maximum total value such that the total size is at most W.



Knapsack problem

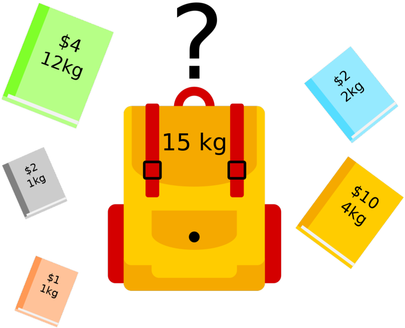
- given a capacity W and n items of values $v_1, \ldots, v_n$ and sizes $w_1, \ldots, w_n$, find a subset of the items having a maximum total value such that the total size is at most W.



Sources: https://en.wikipedia.org and https://twitter.com/

Knapsack problem

- given a capacity W and n items of values $v_1, \ldots, v_n$ and sizes $w_1, \ldots, w_n$, find a subset of the items having a maximum total value such that the total size is at most W.

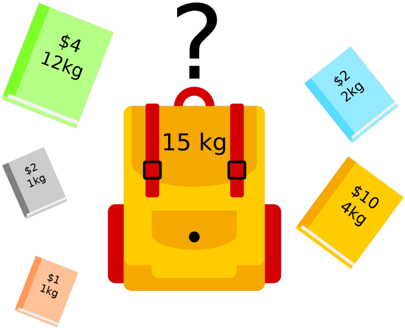


Sources: https://en.wikipedia.org and https://twitter.com/

- This problem is NP-hard.

Knapsack problem

- given a capacity W and n items of values $v_1, \ldots, v_n$ and sizes $w_1, \ldots, w_n$, find a subset of the items having a maximum total value such that the total size is at most W.

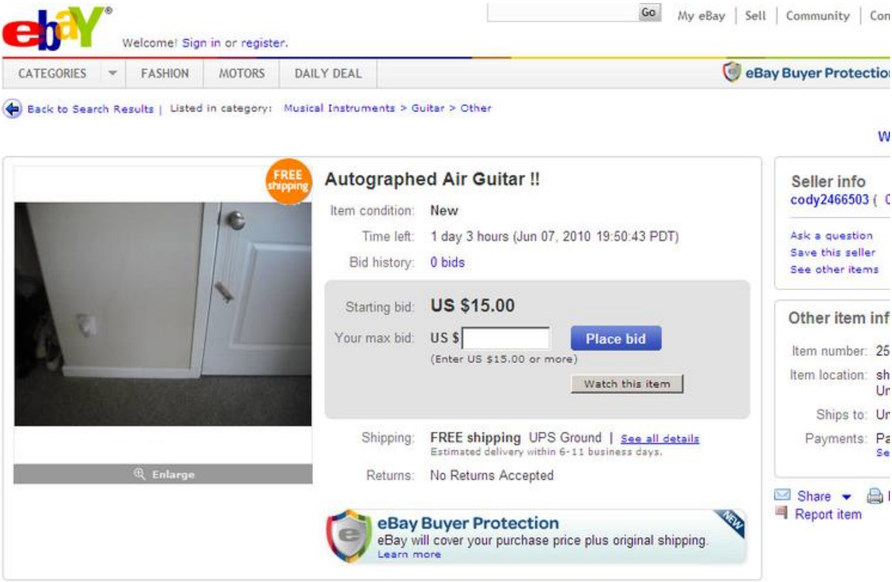


Sources: https://en.wikipedia.org and https://twitter.com/

- This problem is NP-hard.
- There is a pseudo-polynomial time algorithm using dynamic programming and a fully polynomial-time approximation scheme.

- Next lecture, we will focus on revenue maximization.

- Next lecture, we will focus on revenue maximization.



Source: https://goodgearguide.com.au

Source: https://www.kindpng.com/

Thank you for your attention and merry Christmas!