ECONOMICS & COMPUTATION

Spring 2026 | Lecture 9

Strategyproof Approximation Algorithms

Ariel Procaccia | Harvard University

WHEN VCG FALLS SHORT

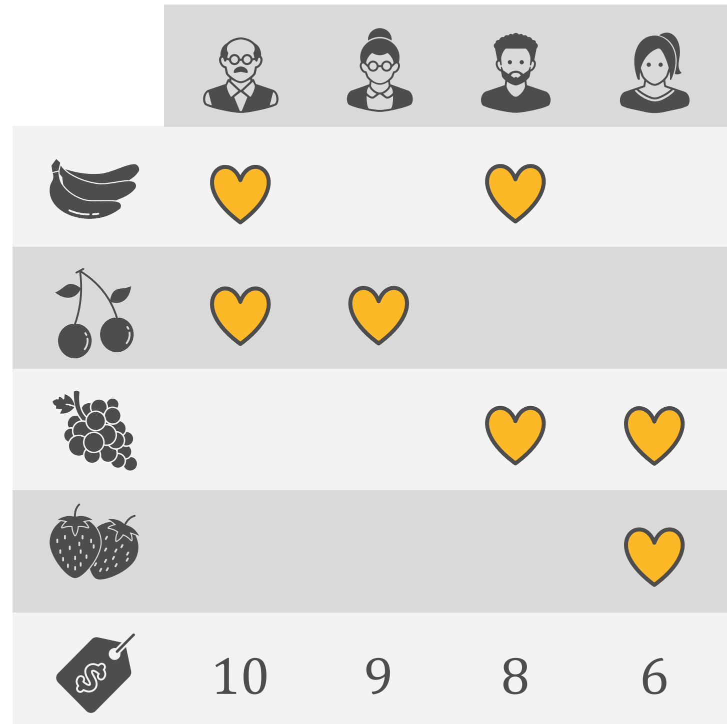
- VCG is an amazing mechanism
- Its Achilles heel, though, is in computing

$$f(\boldsymbol{v}) \in \operatorname{argmax}_{x \in A} \sum_{i \in N} v_i(x)$$

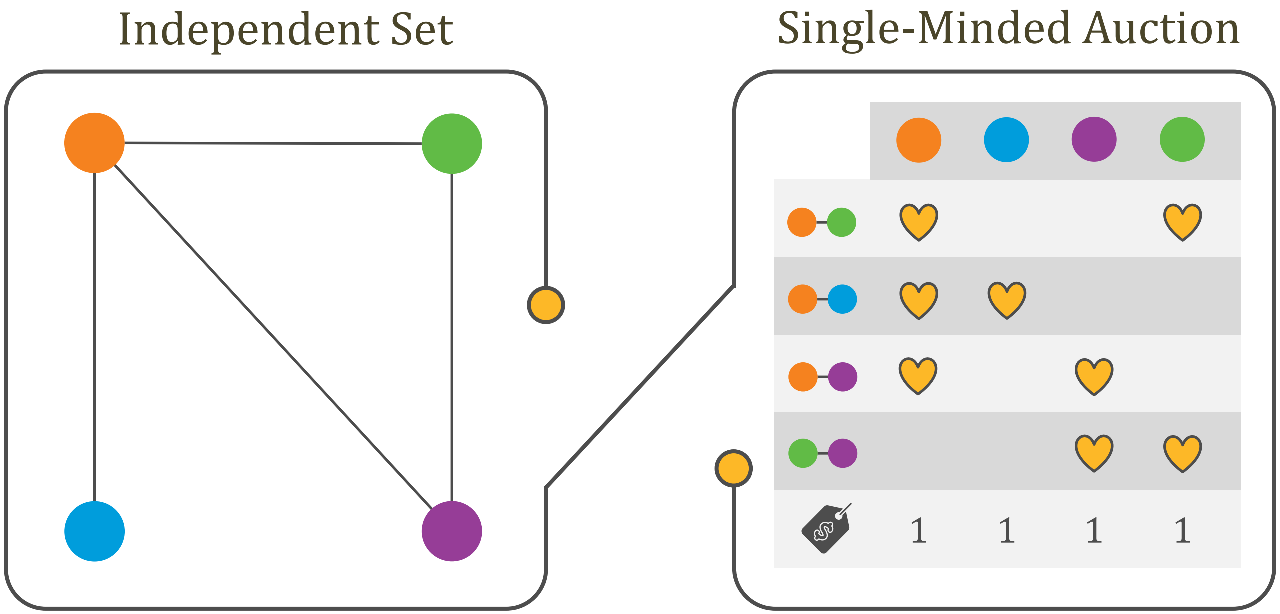
- What do we do if this optimization problem is computationally hard?
- We could solve it approximately, but then we would lose strategyproofness!
- Our goal: approximation and strategyproofness

SINGLE-MINDED AUCTIONS

A set G, $|G| = m$, of goods to allocate. Every player $i \in N$ has a target bundle $T_i \subseteq G$, and has value $v_i(S) = w_i \geq 0$ for $T_i \subseteq S$ and $v_i(S) = 0$ otherwise.



COMPUTATIONAL HARDNESS



- Theorem: Maximizing welfare in single-minded auctions is NP-hard
- Proof:
    - Immediate reduction from Independent Set
    - The set of items is E, there's a player for each vertex, desired bundle is adjacent edges and $w_i = 1$ for all i ■

GREEDY MECHANISM

The greedy single-minded auction for selling a set of items G receives bids (T_i, w_i) for all $i \in N$, and is defined by

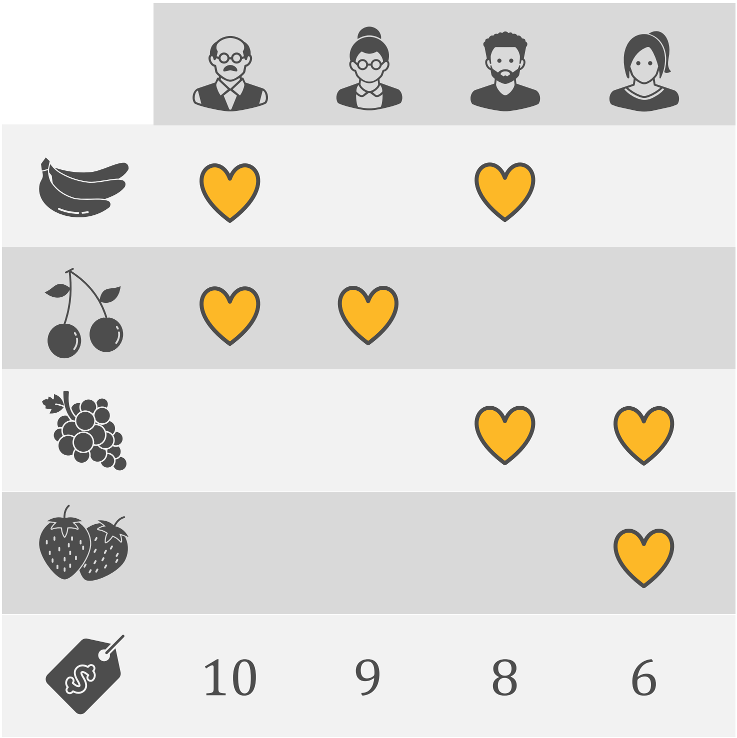
- Allocation rule: sort bids in order of decreasing w_i, breaking ties arbitrarily, and accept bids greedily when they are still feasible
- Payment rule: each allocated player pays the critical value, i.e., the smallest w_i' such that the bid (T_i, w_i') would be accepted (breaking ties for i)

GREEDY MECHANISM: EXAMPLE

Poll 1

What is the payment of the rightmost player?

- 0
- 2
- 3
- 6

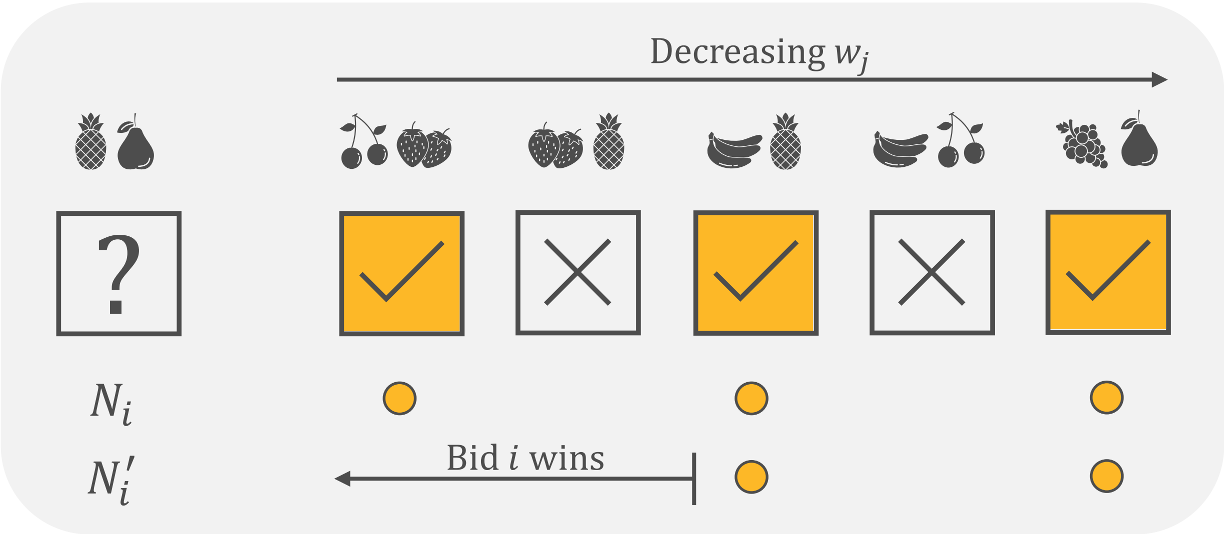


CRITICAL VALUES

- Let N_i be the set of winners if i is removed
- Define the conflict set of i as

$$N_i'(T_i) = \{j \in N_i : T_i \cap T_j \neq \emptyset\}$$

- **Lemma:** Fixing the bids from others, the critical value of i is $w_i^c = \max_{j \in N_i'(T_i)} w_j$



STRATEGYPROOFNESS

- **Theorem:** The greedy single-minded auction is strategyproof
- **Proof:**
  - It isn't useful to report T_i' that doesn't contain T_i, so assume $T_i \subseteq T_i'$
  - By definition, i is allocated T_i' at price $w_i^c(T_i')$ if and only if $w_i' \geq w_i^c(T_i')$
  - It follows (similarly to Vickrey) that $w_i' = w_i$ is optimal
  - (T_i, w_i) is weakly preferred to (T_i', w_i) for any $T_i \subseteq T_i'$ because $w_i^c(T_i) \leq w_i^c(T_i')$ ■

APPROXIMATION

- An algorithm for a maximization problem is a c-approximation algorithm for $c \leq 1$ if for every instance $\mathcal{I}$, $ALG(\mathcal{I}) \geq c \cdot OPT(\mathcal{I})$
- An algorithm for a minimization problem is a c-approximation algorithm for $c \geq 1$ if for every instance $\mathcal{I}$, $ALG(\mathcal{I}) \leq c \cdot OPT(\mathcal{I})$

Poll 2

What is the approximation ratio of the greedy single-minded auction?

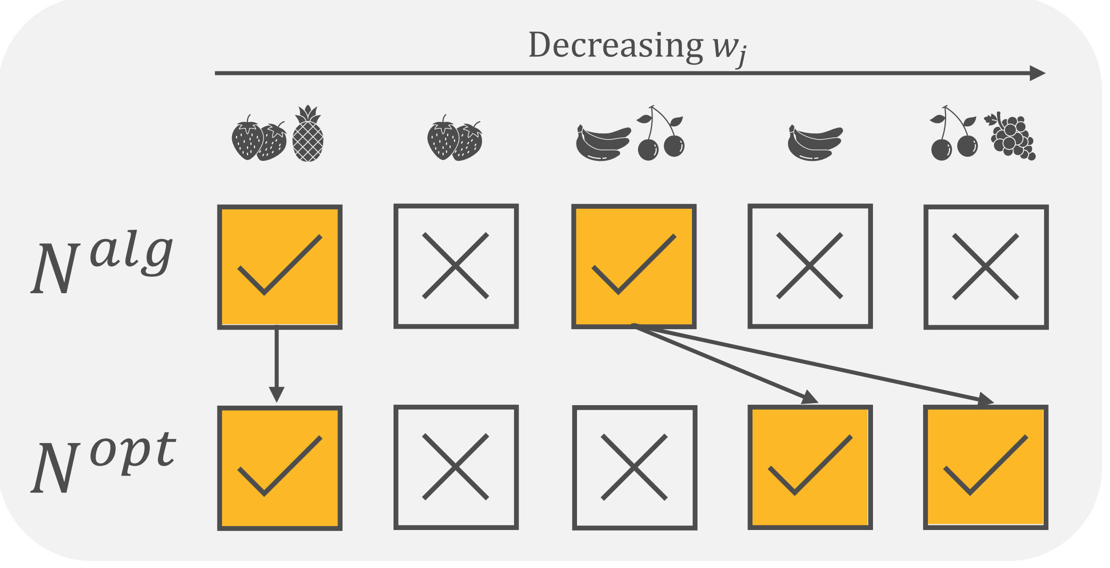
○ $1/2$ ○ $1/3$ ○ $\Theta(1/\log n)$ ○ $\Theta(1/n)$

APPROXIMATION

- Theorem: The greedy single-minded auction is a $1/d$-approximation algorithm, where d is the maximum size of any target bundle
- A variant of the greedy auction where players are ordered by $w_i/\sqrt{|T_i|}$ gives a $1/\sqrt{m}$-approximation for m items
- A better approximation is NP-hard

PROOF OF THEOREM

- Let N^{alg} denote the players allocated under the algorithm, and N^{opt} those allocated under OPT
- For $i \in N^{alg}$, if $i \notin N^{opt}$, let N_i be the set of players $j \in N^{opt}$ such that $w_j \leq w_i$ and $T_i \cap T_j \neq \emptyset$, and if $i \in N^{opt}$, let $N_i = \{i\}$



PROOF OF THEOREM

- It holds that

$$\sum_{j \in N_i} w_j \leq \sum_{j \in N_i} w_i \leq d \cdot w_i$$

- In addition,

$$N^{opt} = \bigcup_{i \in N^{alg}} N_i$$

- We conclude that

$$OPT = \sum_{j \in N^{opt}} w_j \leq \sum_{i \in N^{alg}} \sum_{j \in N_i} w_j \leq d \sum_{i \in N^{alg}} w_i$$
$$= d \cdot ALG \blacksquare$$

TASK ASSIGNMENT

- A set G, $|G| = m$, of tasks to assign
- Player i takes time c_{ij} to complete task j
- An allocation $\boldsymbol{A} = (A_1, \dots, A_n)$ allocates to each player i the tasks in A_i
- The cost of agent i is $\sum_{j \in A_j} c_{ij}$
- The utility of i is $-p_i - \sum_{j \in A_j} c_{ij}$
- The makespan of $\boldsymbol{A}$ is $\max_{i \in N} \sum_{j \in A_j} c_{ij}$, and we want to minimize this objective
- Theorem: Minimizing makespan in task assignment is NP-hard

TASK ASSIGNMENT: EXAMPLE

Instance

	1	10	10	10
	10	1	10	10
	10	1	10	10
	10	2	3	10

Optimal assignment

VCG FOR TASK ASSIGNMENT

- A welfare-maximizing assignment is easy to compute: simply allocate each task j to a player in $\text{argmin}_i c_{ij}$
- Therefore, the VCG mechanism can be implemented in polynomial time!
- But the approximation ratio of VCG for min makespan task assignment is n — why?
- Nisan and Ronen conjectured in 1999 that this is the best approximation ratio a strategyproof (even exponential time) mechanism can have
- In 2023, this conjecture was confirmed by Christodoulou, Koutsoupias and Kovácz