

Algorithmic mechanism design

Vincent Conitzer
conitzer@cs.duke.edu

Algorithmic mechanism design

- Mechanisms should be accompanied by an efficient **algorithm** for computing the outcome
- May not be easy
 - E.g., using the Clarke (VCG) mechanism in combinatorial auctions requires solving the winner determination problem optimally
- If the mechanism's outcomes are too hard to compute, we may need a different mechanism
- **Algorithmic mechanism design** [Nisan & Ronen STOC 99] = simultaneous design of mechanism and algorithm for computing its outcomes
 - Given a mechanism, is there an efficient algorithm for computing its outcomes?
 - Given an algorithm for choosing outcomes, can we make it incentive compatible (e.g., using payments)?

Combinatorial auctions: mechanisms that solve the winner determination problem approximately

- Running Clarke mechanism using approximation algorithms for WDP is generally not strategy-proof
- Assume bidders are single-minded (only want a single bundle)
- A greedy strategy-proof mechanism [Lehmann, O'Callaghan, Shoham JACM 03]:

1. Sort bids by (value/bundle size)	✓ {a}, 11	} $1 \cdot (18/2) = 9$	3. Winning bid pays bundle size times (value/bundle size) of first bid forced out by the winning bid
2. Accept greedily starting from top	✓ {b, c}, 20		
	✗ {a, d}, 18	} $2 \cdot (7/1) = 14$	
	✗ {a, c}, 16		
	✗ {c}, 7		
	✓ {d}, 6	0	

Worst-case
approximation
ratio = $\sqrt{\text{\#items}}$

Can get a better approximation
ratio, $\sqrt{\text{\#items}}$,
by sorting by $\text{value}/\sqrt{\text{bundle
size}}$