2534 Lecture 10: Mechanism Design and Auctions

- Mechanism Design
  - re-introduce mechanisms and mechanism design
  - key results in mechanism design, auctions as an illustration
  - we'll briefly discuss (though we'll likely wrap it up next time):
    - Sandholm and Conitzer's work on *automated mechanism design*
    - Blumrosem, Nisan, Segal: *limited communication auctions*
- Announcements
  - Project proposals back today
  - Assignment 2 in today
  - Projects due on Dec.17

Recap: Second Price Auction

- I want to give away my phone to person values it most
  - in other words, I want to maximize social welfare
  - but I don't know valuations, so I decide to ask and see who's willing to pay: use 2nd-price auction format
- Bidders submit "sealed" bids; highest bidder wins, pays price bid by *second-highest bidder*
  - also known as *Vickrey auctions*
  - special case of *Groves mechanisms*, *Vickrey-Clarke-Groves (VCG) mechanisms*
- 2nd-price seems weird but is quite remarkable
  - truthful bidding, i.e., bidding your true value, is a *dominant strategy*
- To see this, let's formulate it as a Bayesian game

Recap: SPA as a Bayesian Game

- *n* players (bidders)
- Types: each player *k* has *value* $v_k \in [0,1]$ for item
- strategies/actions for player *k*: any bid b_k between [0,1]
- outcomes: player *k* wins, pays price *p* (2nd highest bid)
  - *outcomes are pairs (k,p), i.e., (winner, price)*
- payoff for player *k*:
  - if *k* loses: payoff is 0
  - if *k* wins, payoff depends on price p: payoff is $v_k - p$
- Prior: joint distribution over values (will not specify for now)
  - we do assume that values (types) are *independent and private*
  - i.e., own value does not influence beliefs about value of other bidders
- Note: action space and type space are continuous

Recap: Truthful Bidding: A DSE

- Needn't specify prior: even without knowing others' payoffs, bidding true valuation is *dominant* for every *k*
  - strategy depends on valuation: but *k* selects b_k equal to v_k

- Not hard to see deviation from *truthful bid* can't help (and could harm) *k*, regardless of what others do

- We'll consider two cases: if *k* wins with truthful bid $b_k = v_k$ and if *k* loses with truthful bid $b_k = v_k$

Recap: Equilibrium in SPA Game

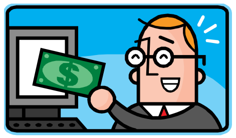
- Suppose *k* wins with truthful bid v_k
  - Notice *k*'s payoff must be positive (or zero if tied)
- Bidding b_k higher than v_k:
  - v_k already highest bid, so *k* still wins and still pays price *p* equal to second-highest bid $b_{(2)}$
- Bidding b_k lower than v_k:
  - If b_k remains higher than second-highest bid $b_{(2)}$ no change in winning status or price
  - If b_k falls below second-highest bid $b_{(2)}$ k now loses and is worse off, or at least no better (payoff is zero)

Recap: Equilibrium in SPA Game

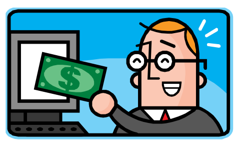
- Suppose k loses with truthful bid v_k
  - Notice k's payoff must be zero and highest bid $b_{(1)} > v_k$
- Bidding b_k lower than v_k:
  - v_k already a losing bid, so k still loses and gets payoff zero
- Bidding b_k higher than v_k:
  - If b_k remains lower than highest bid $b_{(1)}$, no change in winning status (k still loses)
  - If b_k is above highest bid $b_{(1)}$, k now wins, but pays price p equal to $b_{(1)} > v_k$ (payoff is negative since price is more than it's value)

- So a truthful bid is *dominant*: optimal no matter what others are bidding

Truthful Bidding in Second-Price Auction

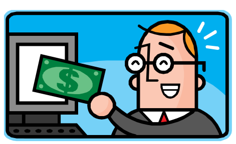


b_1 = \$125



v_2 = \$105

b_2 = ???

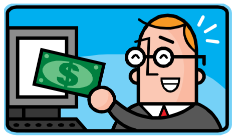


b_3 = \$90

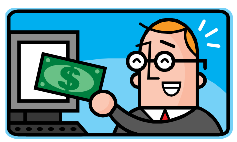
b_4 = \$65

- Consider actions of bidder 2
  - Ignore values of other bidders, consider only their bids. Their values don't impact outcome, only bids do.
- What if bidder 2 bids:
  - truthfully \$105?
    - loses (payoff 0)
  - too high: \$120
    - loses (payoff 0)
  - too high: \$130
    - wins (payoff -20)
  - too low: \$70
    - loses (payoff 0)

Truthful Bidding in Second-Price Auction

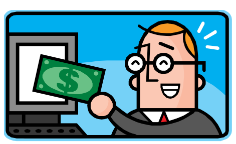


$b_1 = \$95$



$v_2 = \$105$

$b_2 = ???$



$b_3 = \$90$

$b_4 = \$65$

- Consider actions of bidder 2
  - Ignore values of other bidders, consider only their bids. Their values don't impact outcome, only bids do.
- What if bidder 2 bids:
  - truthfully $105?
    - wins (payoff 10)
  - too high: $120
    - wins (payoff 10)
  - too low: $98
    - wins (payoff 10)
  - too low: $90
    - loses (payoff 0)

Other Properties: Second-Price Auction

- Elicits true values (payoffs) from players in game even though they were unknown a priori
- Allocates item to bidder with highest value (maximizes social welfare)
- Surplus is divided between seller and winning buyer
  - splits based on second-highest bid (this is the lowest price the winner could reasonably expect to pay)
- Outcome is similar to Japanese/English auction (ascending auction)
  - consider process of raising prices, bidders dropping out, until one bidder remains
  - until price exceeds *k*'s value, *k* should stay in auction
    - drop out too soon: you lose when you might have won
    - drop out too late: will pay too much if you win
  - last bidder remaining has highest value, pays 2nd highest value! (with some slop due to bid increment)

Mechanism Design

- SPA offers a different perspective on use of game theory
  - instead of predicting how agents will act, we *design* a game to facilitate interaction between players
  - aim is to ensure a *desirable outcome* assuming agents act rationally
- This is the aim of *mechanism design (implementation theory)*
- Examples:
  - voting/policy decisions: want policy preferred by majority of constituents
  - resource allocation/usage: want to assign resources for maximal societal benefit (or maximal benefit to subgroup, or …); often includes determination of payments (e.g., "fair" or "revenue maximizing" or …)
  - task distribution: want to allocate tasks fairly (relative to current workload), or in a way that ensures efficient completion, or …
- Recurring theme: we usually don't know the preferences (payoffs) of society (participants): hence Bayesian games
  - and often incentive to keep these preferences hidden (see examples)

Mechanism Design: Basic Setup

- Set of possible *outcomes* O
- n players, with each player k having:
  - *type space* Θ_k
  - utility function $u_k : O \times \Theta_k \to \mathbf{R}$
    - $u_k(o, \theta_k)$ is utility of outcome o to agent k when type is $\theta_k \in \Theta_k$
    - think of θ_k as an encoding of k's preferences (or utility function)
- (Typically) a common prior distribution P over Θ
- A *social choice function (SCF)* C: $\Theta \to O$
  - intuitively $C(\theta)$ is the most desirable option if player preferences are θ
  - can allow "correspondence", social "objectives" that score outcomes
- Examples of social choice criteria:
  - make majority "happy"; maximize social welfare (SWM); find "fairest" outcome; make one person as happy as possible (e.g., revenue max'ztn in auctions), make least well-off person as happy as possible…
  - set up for SPA: types: values; outcomes: winner-price; SCF: SWM

A Mechanism

- A *mechanism* $((A_k),M)$ consists of:
  - $(A_1,\ldots, A_n)$: *action (strategy) sets* (one per player)
  - an *outcome function* $M: A \longrightarrow \Delta(O)$ (or $M: A \longrightarrow O$)
  - intuitively, players given actions to choose from; based on choice, outcome is selected (stochastically or deterministically)
  - for many mechanisms, we'll break up outcomes into core outcome plus monetary transfer (but for now, glom together)
- Second-price auction:
  - A_k is the set of bids: [0,1]
  - M selects winner-price in obvious way
- Given a mechanism design setup (players, types, utility functions, prior), the mechanism induces a *Bayesian game* in the obvious way

Implementation

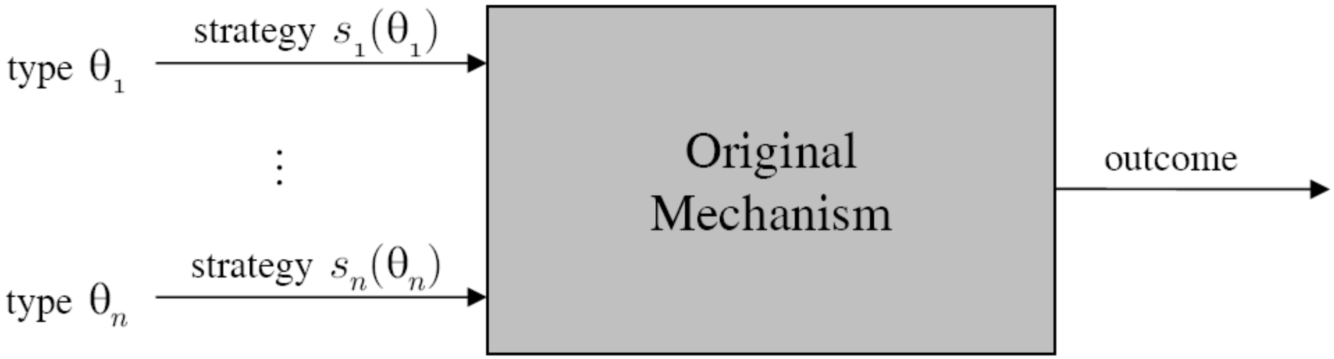
- What makes a mechanism useful?
  - it should implement the social choice function *C*
  - i.e., if agents act "rationally" in the Bayesian game, outcome proposed by *C* will result
  - of course, rationality depends on the equilibrium concept
- A mechanism *(A,M)* *S-implements* *C* iff for (some/all) S-solutions σ of the induced Bayesian game we have, for any $\theta \in \Theta$, $M(\sigma(\theta)) = C(\theta)$
  - here S may refer to DSE, ex post equilibrium, or Bayes-Nash equilibrium
  - in other words, when agents play an equilibrium in the induced game, whenever the type profile is θ, then the game will give the same outcome as prescribed for θ by the social choice function
  - notice some indeterminacy (in case of multiple equilibria)
- For SCF *C* = "maximize social welfare" (including seller as a player, and assuming additive utility in price/value), the SPA implements SCF in dominant strategies

Revelation Principle

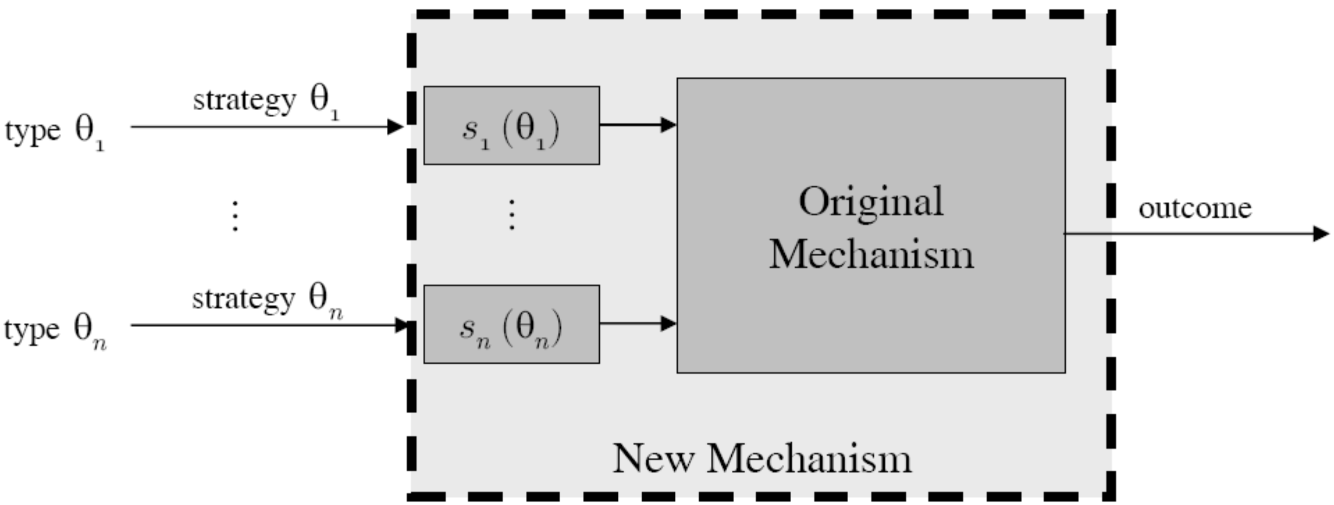
- Given SCF *C*, how could one even begin to explore space of mechanisms?
  - actions can be arbitrary, mappings can be arbitrary, …
- Notice that SPA keeps actions simple: "state your value"
  - it's a *direct mechanism:* $A_k = \theta_k$ (i.e., actions are "declare your type")
  - …and stating values truthfully is a DSE
  - Turns out this is an instance of a broad principle
- **Revelation principle:** if there is an S-implementation of SCF *C*, then there exists a direct, mechanism that S-implements *C* and is truthful
  - intuition: design new outcome function *M'* so that when agents report truthfully, the mechanism makes the choice that the original *M* would have realized in the S-solution
- Consequence: much work in mechanism design focuses on direct mechanisms and truthful implementation

Revelation Principle

Fig from Multiagent Systems, Shoham and Leyton-Brown, 2009



(a) Revelation principle: original mechanism



(b) Revelation principle: new mechanism

If truthful reporting not in EQ in New, then some agent k wants an action different than that dictated by s_k under her true type. But this means s_k was not in EQ in Original.

Gibbard-Satterthwaite Theorem

- Dominant strategy implementation a frequent goal
  - agents needn't rely on any strategic reasoning, beliefs about types
  - unfortunately, DS implementation not possible for general SCFs
- **Thm (Gibbard73, Sattherwaite75):** Let C (over N, O) be s.t.:

  (i) $|O| > 2$;

  (ii) C is onto (every outcome is selected for some profile θ);

  (iii) C is non-dictatorial (there is no agent whose preferences "dictate" the outcome, i.e., who always gets max utility outcome);

  (iv) all preferences are possible.

  Then C cannot be implemented in dominant strategies.
- Proof (and result) similar to Arrow's Thm (which we'll see shortly)
- Ways around this:
  - use weaker forms of implementation
  - restrict the setting (especially: consider special classes of preferences)

Groves Mechanisms

- Despite GS theorem, truthful implementation in DS is possible for an important class of problems
  - assume outcomes allow for transfer of utility between players
  - assume agent preferences over such transfers are additive
  - auctions are an example (utility function in SPA)
- *Quasi-linear mechanism design problem (QLMD)*
  - extend outcome space with "monetary" transfers
    - outcomes: $O \times T$, where T is set of vectors of form $(t_1, \ldots t_n)$
  - *quasi-linear utility*: $u_k((o,t),\theta_k) = v_k(o,\theta_k) + t_k$
  - SCF is SWM (i.e., maximization of social welfare $SW(o,t,\theta)$)
- Assumptions:
  - value for "concrete" outcomes is commensurate with transfer
  - players are risk neutral
- *In SPA, utility is valuation less price paid (negative transfer to winner), or price paid (positive transfer to seller)* *(see formalization on slide 3)*

Groves Mechanisms

- *A Groves mechanism (A,M)* for a QLMD problem is:
  - $A_k = \theta_k = V_k$: agent k announces values v^*_k for outcomes
  - $M(v^*) = (o, t_1, \ldots t_n)$ where:
    - $o = argmax_{o \in O} \sum_k v^*_k(o)$
    - $t_k(v^*_k) = \sum_{j \neq k} v^*_j(o) - h_k(v^*_{-k})$, where h_k is an arbitrary function
- Intuition is simple:
  - choose SWM-outcome based on *declared* values v^*
  - then transfer to k: the *declared welfare* of chosen outcome to the *other agents*, less some "social cost" function h_k which depends on what others said *(but critically, not on what k reports)*
- Some notes:
  - in fact, this is a family of mechanisms, for various choices of h_k
  - if agents reveal true values, i.e., $v^*_k = v_k$ for all k, then it maximizes SW
  - SPA: is an instance of this

Truthfulness of Groves

- **Thm:** Any Groves mechanism is truthful in dominant strategies (*strategyproof*) and efficient. Proof easy to see:
  - outcome is: $o = argmax_{o \in O} \sum_k v^*_k(o)$
  - k receives: $t_k(v^*) = \sum_{j \neq k} v^*_j(o) - h_k(v^*_{-k})$
  - k's utility for report v^*_k is: $v_k(o) + \sum_{j \neq k} v^*_j(o) - h_k(v^*_{-k})$,
    - *here o depends on the report* v^*_k
  - k wants to report v^*_k that maximizes $v_k(o) + \sum_{j \neq k} v^*_j(o)$
    - *this is just k's utility plus reported SW of others*
    - *notice k's report has no impact on third term* $h_k(v^*_{-k})$
  - but mechanism chooses o to max reported SW, so no report by k can lead to a better outcome for k than v_k
  - efficiency (SWM) follows immediately
- This is why SPA is truthful (and efficient)

Other Properties of Groves

- Famous theorem of Green and Laffont: The Groves mechanism is unique in the following sense---any mechanism for a QLMD problem that is truthful, efficient is a Groves mechanism (i.e., must have payments of the Groves form)
    - see proof sketch in S&LB
- Famous theorem of Roberts: the only SCFs that can be implemented truthfully (with no restrictions on preferences) are *affine maximizers* (basically, SWM but with weights/biases for different agents' valuations)
- Together, these show the real centrality of Groves mechanisms

Participation in the mechanism

- While agents *participating* will declare truthfully, why would agent participate? What if h_k = -*LARGEVALUE?*
- *Individual rationality:* no agent loses by participating in mechanism
  - basic idea: your expected utility positive (non-negative), i.e., the value of outcome. should be greater than your payment
- *Ex interim IR:* your expected utility is positive for every one of your types/valuations (taking expectation over $Pr(v_{-k} \mid v_k)$):
  - $E [\, v_k(M(\sigma_k(v_k), \sigma_{-k}(v_{-k}))) - t_k(\sigma_k(v_k), \sigma_{-k}(v_{-k}))\,] \geq 0$ for all k, v_k
    - where σ is the (DS, EP, BN) equilibrium strategy profile
- *Ex post IR:* your utility is positive for every type/valuation (even if you learn valuations of others):
  - $v_k(M(\sigma(v))) - t_k(\sigma(v)) \geq 0$ for all k, v
    - where σ is the (DS, EP, BN) equilibrium strategy profile

- Ex ante IR can be defined too (a bit less useful, but has a role in places)

VCG Mechanisms

- *Clarke tax* is a specific social cost function *h*
  - $h_k(v^*_{-k}) = max_{o \in O[-k]} \sum_{j \neq k} v^*_j(o)$
  - assumes subspace of outcomes *O[-k]* that don't involve *k*
  - $h_k(v^*_{-k})$: how well-off others would have been had *k* not participated
  - total transfer is value others received with *k*'s participation less value that *they would have received* without *k* (i.e., "externality" imposed by *k*)
- With Clarke tax, called *Vickrey-Clarke-Groves (VCG) mechanism*

- **Thm:** VCG mechanism is strategyproof, efficient and *ex interim individually rational (IR)*.

- It should be easy to see why SPA (aka Vickrey auction) is a VCG mechanism
  - what is externality winner imposes?
  - valuation of second-highest bidder (who doesn't win because of presence)

Other Issues

- Budget balance: transfers sum to zero
  - transfers in VCG need not be balanced (might be OK to run a surplus; but mechanism may need to subsidize its operation)
  - general impossibility result: if type space is rich enough (all valuations over O), can't generally attain efficiency, strategyproofness, and budget balance
  - some special cases can be achieved (e.g., see "no single-agent effect", which is why VCG works for very general single-sided auctions), or the dAGVA mechanism (BNE, ex ante IR, budget-balanced)
- Implementing other choice functions
  - we'll see this when we discuss social choice (e.g., maxmin fairness)
- Ex post or BN implementation
  - e.g., the dAGVA mechanism

Issues with VCG

- *Type revelation*
  - revealing utility functions difficult; e.g., large (combinatorial) outcomes
    - privacy, communication complexity, computation
  - can incremental elicitation work?
    - sometimes: e.g., descending (Dutch auction)
  - can approximation work?
    - in general, no; but sometime yes… we'll discuss more in a bit…
- *Computational approximation*
  - VCG requires computing optimal (SWM) outcomes
    - not just one optimization, but *n+1* (for all *n* "subeconomies")
    - often problematic (e.g., combinatorial auctions)
    - focus of algorithmic mechanism design
  - But approximation can destroy incentives and other properties of VCG

Issues with VCG

- Frugality
  - VCG transfers may be more extreme than seems necessary
    - e.g., seller revenue, total cost to buyer
    - we'll see an example in combinatorial auctions
  - a fair amount of study on design of mechanisms that are "frugal" (e.g., that try to minimize cost to a buyer) in specific settings (e.g., network and graph problems)
- Collusion
  - many mechanisms are susceptible to collusion, but VCG is largely viewed as being especially susceptible (we'll return to this: auctions)
- Returning revenue to agents
  - an issue studied to some extent: if VCG extracts payments over and above true costs (e.g., Clarke tax for public projects), can some of this be returned to bidders (in a way that doesn't impact truthfulness)?

Combinatorial Auctions

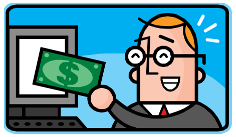
- Already discussed 2nd price auctions in depth, 1st price auctions a bit (and will return in a few slides to auctions in general)

- Often sellers offer multiple (distinct) items, buyers need multiple items
  - buyer's value may depend on the collection of items obtained
- *Complements:* items whose value increase when combined
  - e.g., a cheap flight to Siena less valuable if you don't have a hotel room
- *Substitutes:* items whose value decrease when combined
  - e.g., you'd like the 10AM flight or the 7AM flight; but not both
- If items are sold separately, knowing how to bid is difficult
  - bidders run an *"exposure" risk:* might win item whose value is unpredictable because unsure of what other items they might win

Simultaneous Auctions: Substitutes

Flight1 (7AM, no airmiles, 1 stopover)
Value: $750

Flight2 (10AM, get airmiles, direct)
Value: $950



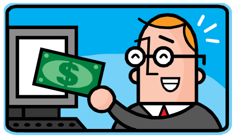
Bidder can only use ***one*** of the flights:
Value of receiving both flights is $950

- If both flights auctioned *simultaneously*, how should he bid?
- Bid for both? runs the risk of winning both (and would need to hedge against that risk by underbidding, reducing odds of winning either)
- Bid for one? runs the risk of losing the flight he bids for, and he might have won the other had he bid
- If items auctioned *in sequence*, it can mitigate risk a bit; but still difficult to determine how much to bid first time

Simultaneous Auctions: Complements

Flight1

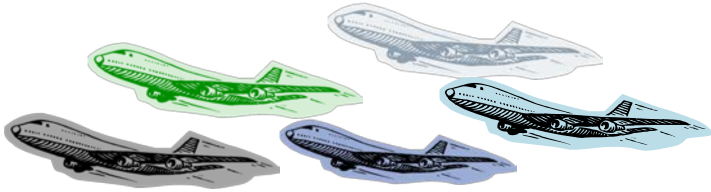
Hotel Room

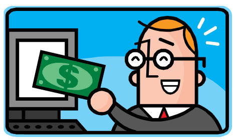


Bidder doesn't want flight without hotel room,
or hotel without flight; but together, value is $1250

- If flight, hotel auctioned *simultaneously*, how should he bid?
- Useless to bid for only one; but if he bids for both, he runs the risk of winning only one (which is worthless in isolation). Requires severe hedging/underbidding to account for this risk.
- If items auctioned *in sequence*, it can mitigates risk only a little bit. If he loses first item, fine. If he wins, will need to bid very aggressively in second (first item a “sunk cost”) and can end up overpaying for pair

Combinatorial Auction





Bidder expresses value for ***combinations*** of items:

- Value(flight2, hotel1) = $1250
- Value(flight1, hotel1) = $1050
- Don't want any other package

- *Combinatorial auctions* allow bidders to express *package bids*
  - for any combination of items can say what you are willing to pay for that combination or package
  - do not pay unless you get exactly that package
  - outcome of auction: assign (at most) one package to each bidder
  - can use 1st-price (pay what you bid) or VCG

Combinatorial and Expressive Auctions

- *Expressive bidding* in auctions becoming common
  - expressive languages allow: combinatorial bids, side-constraints, discount schedules, etc.
  - direct expression of utility/cost: economic efficiency
- Advances in *winner determination*
  - determine least-cost allocation of business to bidders
  - new optimization methods key to acceptance
  - applied to large-scale problems (e.g., sourcing)

Reverse Combinatorial Auctions

- Buyer: desires collection of items *G*
- Sellers: offer "bundle" bids $\langle b_i, p_i \rangle$, where $b \subseteq G$
  - possibly side constraints (seller, buyer)
- *Feasible allocation*: subset $B' \subseteq B$ covering *G*
  - let *X* denote the set of feasible allocations
- Winner determination: find the least-cost allocation
  - formulate this as an integer program
    - variable q_i indicates acceptance of bid b_i

$$\min p_i q_i \;\; s.t. \sum_{i:g \in b_i} q_i \geq 1, \; \forall g \in G$$

  - can add all sorts of side constraints, discounts, etc.
  - NP-hard, inapproximable, but lots of research on "practically effective" algorithms, special cases, …

Incentives in Combinatorial Auctions

- How could you get bidders to reveal their true costs?
- Use VCG
  - collect bundle bids $\langle b_k, p_k \rangle$ from each bidder
  - find optimal allocation *a* (min cost set of bundles covering requirements): has cost *c*
  - for each winning (accepted) bidder *k*, compute the optimal allocation without his bid: has higher cost c_k
  - accept bids in optimal allocation *a*, and pay (receive from) each winning bidder using VCG: $b_k + (c_k - c)$

Potential Problems with VCG for CAs

- Winner determination is NP-complete and inapproximable
  - yet we don't just solve it once, we solve it m times (m winning bidders)
  - in practice, VCG is seldom used in CAs
  - sealed-bid: uses first-pricing; but ascending auctions sometimes used which can have VCG-like properties
- It would be nice to use an approximation algorithm
  - but truthfulness and IR guarantees go away (in practice, not a problem)
- Can overpay severely (*reverse* auction example, Conitzer-Sandholm)
  - n items: two bidders offer to supply all n, A at price p, B at price $q < p$
    - B wins and is paid $p = q + (p - q)$
  - now add n bidders $C_1 \ldots C_n$, each offering one good for free
    - the C's win and are paid q each: total payment is $n*q$
    - *adding bidders increased the total price paid significantly (and not frugal with respect to true cost)*
    - *note also how susceptible to collusion*

Auctions

- Auctions widely used (to both sell, buy things)
  - our SPA was a *one-sided, sell-side* auctions: that is, we have a single seller, and multiple buyers
  - examples: rights to use public resources (timber, mineral, oil, wireless spectrum), fine art/collectibles, Ebay, online ads (Google, Yahoo!, Microsoft, …), …
- Variations:
  - *multi-item auctions:* one seller, multiple items at once
    - e.g., wireless spectrum, online ads
    - interesting due to substitution, complementarities (see CAs)
  - *procurement (reverse) auctions:* one buyer, multiple sellers
    - common in business for dealing with suppliers
    - government contracts tendered this way
    - aim: purchase items from *cheapest* bidder (meeting requirements)
  - *double-sided auctions:* multiple sellers and buyers
    - stock markets a prime example, *matching* is the critical problem

Single-item Auctions (Sell-side)

- Assume seller with one item for sale
- Several different formats
  - **Ascending-bid (open-cry) auctions** (aka *English auctions*)
    - price rises over time, bidders drop out when price exceeds their “comfort level”; final bidder left wins item at last drop-out price
  - **Descending-bid (open-cry) auctions** (aka *Dutch auctions*)
    - price drops over time, bidders indicate willingness to buy when price drops to their “comfort level”; first bidder to indicate willingness to buy wins at that price
  - **First-price (sealed bid) auctions**
    - bidders submit “private” bids; highest bidder wins, pays price he bid
  - **Second-price (sealed bid) auctions**
    - bidders submit “private” bids; highest bidder wins, pays price bid by the second-highest bidder

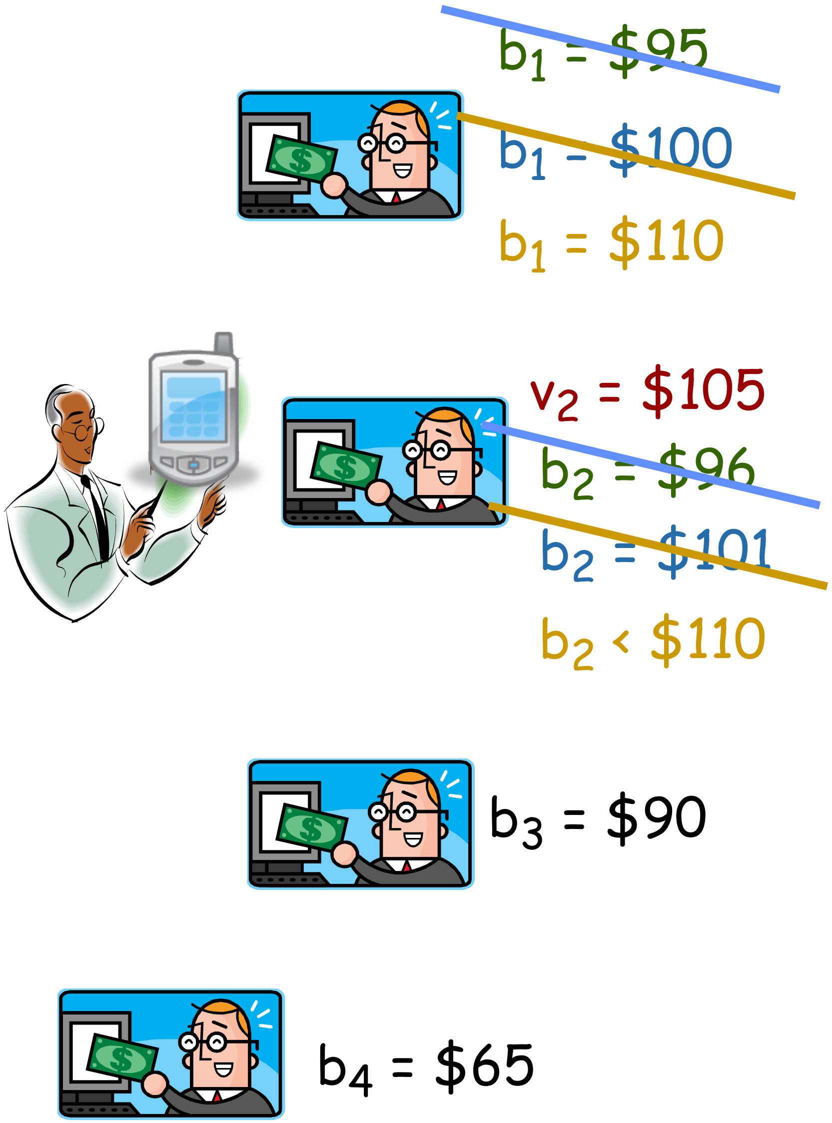
The First-Price Auction Game

- *n* players (bidders)
- Types: each player *k* has *value* $v_k \in [0,1]$ for item
- Prior: assume all valuations are distributed uniformly on [0,1]
  - unlike SPA, prior will be critical here (of course, other priors possible)
- strategies/actions for player *k*: any bid b_k between [0,1]
- outcomes: player *k* wins, pays price *p* (her own highest bid)
  - *outcomes are pairs (k,p), i.e., (winner, price)*
- payoff for player *k*:
  - if *k* loses: payoff is 0
  - if *k* wins, payoff depends on price *p*: payoff is $v_k - p$
- Like SPA, the FPA mechanism induces a Bayesian game among the bidders

First-Price Auction: No dominant strategy

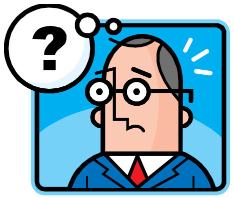
- Notice that there is no dominant strategy for any bidder *k*
- Suppose other players bid: highest bid from others is $b_{(1)}$
  - If value v_k is greater than $b_{(1)}$ then *k*'s best bid is b_k that is just a "shade" greater than $b_{(1)}$ (depends on how ties are broken)
  - This gives *k* a payoff of (just shade under) $v_k - b_{(1)} > 0$
  - If *k* bids less than $b_{(1)}$: *k* loses item (payoff 0)
  - If *k* bids more than $b_{(1)}$: pays more than necessary (so *k*'s payoff is reduced)
  - Notice *k* should never bid more than v_k
- So *k*'s optimal bid depends on what others do
- Thus *k* needs some prediction of how others will bid
  - requires genuine *equilibrium analysis in the Bayes-Nash sense*
  - must predict others' strategies *(mapping from types to bid)* and use beliefs about others' types *(to predict actual bids)*

Bid Shading in First-Price Auction

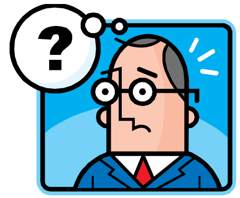


- Consider actions of bidder 2
  - ignore values of other bidders, consider only bids.
  - assume "bid increment" $1and that ties broken against bidder 2
- If bidder 1 bids $95:
  - bidder 2 *should bid* $96
    - wins (payoff 9)
  - if 2 bids $94, loses (0)
  - if 2 bids $97, payoff 8
- If bidder 1 bids $100
  - bidder 2 *should bid* $101
    - wins (payoff 4)
- If bidder 1 bids $110
  - bidder 2 should bid "less"
    - loses (payoff 0)

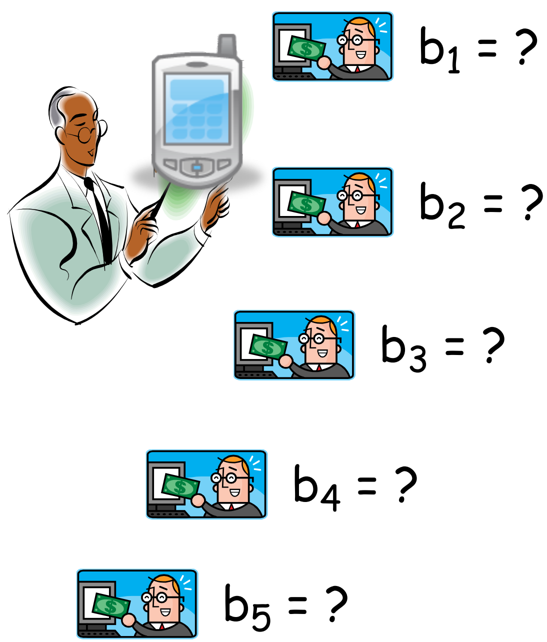
Bid Shading in First-Price Auction

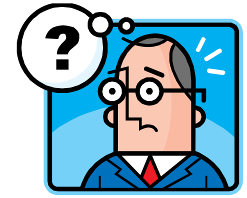


What bid b_k should bidder k offer?

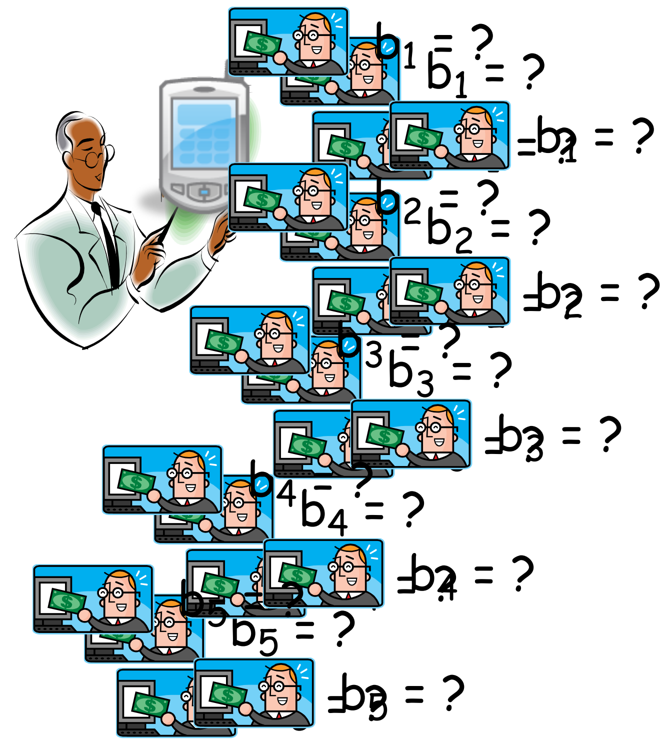


What bid b_k should bidder k offer?





What bid b_k should bidder k offer?



Equilibrium: First-Price Auction

- Let's run through simple analysis
- Game of incomplete information
  - *k*'s *strategy s* depends on value v_k : $s_k(v_k)$ selects a bid b_k in [0,1]
    - other players have strategies too: s_j
  - *k*'s payoff depends on its strategy and the strategy of others (as in Nash equilibrium), but also on *its value* and *the value of others*
    - *i.e., it's a "true" Bayesian game: priors influence bids*
- Let's look at game with two bidders *k* and *j*
  - Assume that their values are drawn randomly (uniformly) from the interval [0,1] and that they both know this
  - Let's see what strategies are in equilibrium…

BNE: 2-bidder 1st Price Auction

- *Bidding strategy* for k : function $s_k(v_k) = b_k$:
  - it tells you what bid to submit taking your value for the item as input
  - e.g., truthful strategy: $s(0)=0$; $s(0.1) = 0.1$; $s(1) = 1$; etc…
  - e.g., $s(v) = ½v$ says "bid half your value": $s(0)=0$; $s(0.1)=0.05$; $s(1) = 0.5$; …
- Some simplifying assumptions
  - strategy is *strictly increasing* (if value is higher, bid is also higher)
    - intuitively makes sense, but some sensible strategies might not
  - strategy is *differentiable*
    - makes analysis easier, but not a critical in general
  - strategy *cannot bid higher than value:* $s(v) \leq v$
    - an obvious requirement for rational bidders
  - strategies are *symmetric*: k and j use same function, s_k same as s_j
    - not necessary: we derive only a symmetric equilibrium (non-symmetric equilibria may also exist)

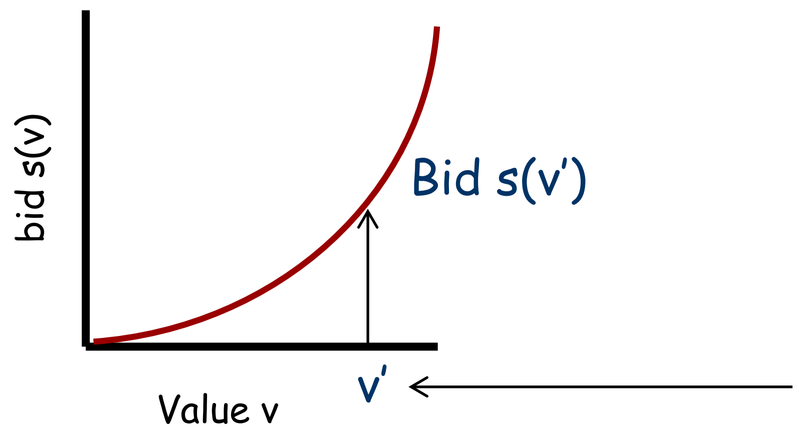
BNE: 2-bidder 1st Price Auction

- By symmetric assumption, *k* never wants to bid more than *s(1)* (since this is the maximum *j* will bid)
  - and obviously *s(0) = 0*, so *k* can't bid less than *s(0)*
- We want to find a strategy *s* such that neither *k* nor *j* deviate from *s*
- But for any strategy *s* satisfying our assumptions (specifically, differentiability), *k* can produce any bid b_k between *s(0)* and *s(1)* by plugging in some "pretend" valuation *v* (possibly different from true v_k)
  - like an internal version of the revelation principle

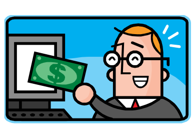
- So we can focus attention (reduce our search) to strategies where the payoff for bidding $s(v_k)$, when *k*'s true value is v_k, is greater than the payoff for bidding *s(v)* for a *different* value *v* when *k*'s true value is v_k

Fixing a strategy and changing the bid

- Even with a fixed strategy *s*, bidder *k* can produce any bid between *0* and *s(1)* by “pretending” to have a different value *v’* than his true *v*
  - … and it’s his bid that influences the outcome, not *s* per se



Plug in any value v’ you want (“lie to yourself”) to get any desired bid between 0 and s(1)



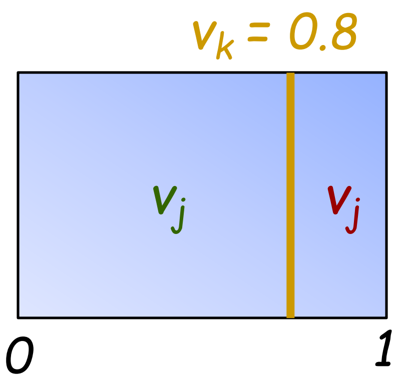
Bidder k

What is expected value of strategy *s*?

- What is *k*'s expected payoff for playing *s*?
  - Payoff is zero if *k* loses
  - Payoff is "value minus bid" if *k* wins: $v_k - s(v_k)$
  - So if *k* wins with probability *p*, expected payoff is $p(v_k - s(v_k))$
- What is probability *k* wins?
  - Since strategies are symmetric, *k* wins just when $v_k > v_j$
  - This happens with probability v_k
  - So *k*'s expected payoff is $v_k(v_k - s(v_k))$

$Prob(v_j < 0.8) = 0.8$

$Prob(v_j > 0.8) = 0.2$



What is optimal bidding strategy?

- Want a strategy *s* where expected value of *bidding true valuation* v_k is better than bidding any other valuation *v*
  - If true valuation is v_k and bid is *v*: probability of winning is *v*, and payoff if bidder wins is $v_k - s(v)$
  - So we want *s* satisfying: $v_k(v_k - s(v_k)) \geq v(v_k - s(v))$ for all *v*
  - i.e., payoff function $g(v) = v(v_k - s(v))$ must be maximized by input v_k

$$g'(v_k) = 0$$

$$v_k - s(v_k) - v_k s'(v_k) = 0$$

$$s'(v_k) = 1 - \frac{s(v_k)}{v_k}$$

- Result is: $s(v) = v/2$
- In other words, *the bidding strategy where both bidders bid half of their valuation is a Nash equilibrium*

For More Than Two Bidders

- Same analysis can be applied (uniform valuations on any bounded interval) to give an intuitive result:
- If we have *n* bidders, the (unique) symmetric equilibrium strategy is for any bidder with valuation v_i to bid $(n\text{-}1)/n\ v_i$
  - e.g., if 2 bidders, bid half of your value
  - e.g, if 10 bidders, bid 9/10 of your value
  - e.g, if 100 bidders, bid 99% of your value

- Each bidder: bids expectation of highest valuation excluding his own (conditioned on his valuation being highest)
- Intuition (again): more competing bidders means that there is a greater chance for higher bids: so you sacrifice some payoff ($v_i - b_i$) to increase probability of winning in a more “competitive” situation

Symmetric Equilibria in General

- Analysis more involved for general CDF *F* over valuations
  - each specific form requires its own analysis, but general picture is very similar to the uniform distribution case
- Still, general principle holds in symmetric equilibrium:

$$s(v_k) = E_{V\sim F}[\, V_{(1)} \mid V_{(1)} < v_k \,],$$

where $V_{(1)}$ is the highest value of $n\text{-}1$ independent draws from F

Other Properties: First-Price Auction

- Bidders generally shade bids (as we've seen)
  - Does seller lose revenue compared to second-price auction?
- If bidders all use same (increasing) strategy, item goes to bidder with highest value (will maximizes social welfare, like second-price)
  - but note that our symmetric equilibrium needn't be only one
- Outcome is similar to Dutch auction (descending auction)
  - lower prices until one bidder accepts the announced price
  - until price drops below *k*'s value, *k* should not accept it
    - jump in too soon: will pay more than necessary (equivalent to bid shading)
    - jump in too late: you lose when you might have won
  - first bidder jumping in pays the price she jumped in at (1^{st} price)
  - games are in fact "strategically equivalent"; seller gets same price
    - with some "slop" due to bid decrement in Dutch auction

Revenue Equivalence

- Goal of auction may be to maximize revenue to seller
  - this is just a different SCF
  - do any of these auctions vary in expected revenue?
- First note that 1st and 2nd price net same expected revenue: expectation of $v_{(2)}$
- Revenue equivalence
  - under a set of reasonable assumptions, all auctions (assuming symmetric equilibrium play) result in a bidder with a specific valuation v_k making the *same expected payment*, hence lead to the *same expected revenue* for the seller
  - assumptions: IPV from bounded interval $[v_{low}, v_{high}]$, F is strictly increasing (atomless), auction is efficient, bidder with v_{low} has expected utility (hence payment) zero

Reserve Prices and Optimal Auctions

- If SCF is revenue maximization, none of the auction formats implement this SCF
- Well-chosen *reserve price r* increases revenue to seller
  - reserve prices also make sense when seller has value for item
- In 2nd price (notice still dominant to bid truthfully):
  - runs risks of not selling item (all bids below *r*)
  - increases sale price if $v_{(1)} > r > v_{(2)}$
  - no impact *if* $v_{(2)} > r$
- In 1st price: bid "as before:" $E[max(r,V_{(1)}) | V_{(1)} < v_k]$
- Revenue improves if *r* set carefully to balance *probability of not selling* against *increased price when item is sold*
- A rather simple optimization, but relies on CDF *F* over valuations
  - hence used rarely in practice (but see discussion of AMD)

Optimal Reserve Price

- Suppose IPV, prior density *f* (with CDF *F*) over valuations
  - let *g* be density (with CDF *G*) over highest value from *n-1* draws from *f*
- Expected payment (1st or 2nd price auction) of bidder *k* with val v_k :
  - ***If*** *k* wins: pays *r* if 2nd highest val less than *r*; 2nd highest val otherwise

$$rG(r) + \int_r^{v_k} y g(y) dy$$

- Pay r with $\Pr(v_{(2)} < r)$
- Pay y>r with $\Pr(v_{(2)} = y)$

- Ex ante expected payment is then:

$$r(1 - F(r))G(r) + \int_r^{v_{high}} y(1 - F(y))g(y) dy$$

- Pay r: $\Pr(v_{(2)} < r) * \Pr(v_k \geq r)$
- Pay y>r: $\Pr(v_{(2)} = y) * \Pr(v_k \geq y)$

- Expected revenue to seller is *n* times this (*n* bidders)
- Optimal reserve price *r** should satisfy (w/ mild assumptions of F, f):

$$r^* - \frac{1 - F(r^*)}{f(r^*)} = 0$$

Myerson Auction

- Myerson auction generalizes these insights, allowing for knowledge of each bidder's "personal" CDF F_k
  - Does some bid shading for the bidder and sets "personalized reserve prices" for each bidder
  - Bidder submits valuation v_k
  - Compute virtual valuation ψ_k
  - Set reserve price r_k satisfying $\psi_k(r_k) = 0$
  - Award item to bidder k^* with highest virtual valuation (if above reserve)
  - Price p = smallest valuation that would have still allowed k* to win

$$\psi_k(v_k) = v_k - \frac{1 - F_k(v_k)}{f_k(v_k)}$$

- Properties
  - Bidding truthfully still dominant
  - Can awards item to bidder with lower valuations (but higher virtual valuation): increases power of bidders with lower true valuations to put pressure on bidders with higher valuations (increases competition)
  - Provably maximizes seller revenue

Common/Correlated Values

- Five companies bidding (1st-price) for oil drilling rights in area *A*
  - ultimate value is pretty much the same for each: a certain amount of oil *(B bbls);* each will sell it at market price (ignore technology differences)
  - seller, companies don't know the value
  - each produces its own *(private) estimate* of the reserves (quantity *B*)
    - value is now *random (probabilistic):* bid based on your *expected value*
- Estimates are related to *B*, but noisy (error-prone):
  - e.g., *U* estimates 50M bbl; V: 47M; W: 42M; X: 40M; Y: 38M
  - once *U* wins, learns something about other's estimates: all lower than *U*'s
  - suggests *U*'s estimate was too high: perhaps *U* overpaid!
- Phenomenon is known as *winner's curse*
  - winning auction: implies value is less than you estimated
  - may still profit (attain a surplus), but could even have negative (expected) surplus!
  - occurs in any common/correlated value auction (e.g., buying items for resale)
- Bidding strategies must reflect this (and interesting information flow)

Automated Mechanism Design

- General view in MD
  - hand-designed mechanisms proven to work for wide-class of problems
  - prior independent (VCG), parameterized (Myerson, dAGVA), …
- Drawbacks
  - Gibbard-Satterthwaite: settings are still restrictive
  - specific SCFs, specific preferences (quasi-linearity), etc…
- *Automated mechanism design* [Conitzer and Sandholm]
  - hard work to handcraft mechanisms, so need these to be broad
  - but this generality runs smack into impossibilities (GS, Roberts, etc.)
  - if you have specific info about problem at hand, generality not needed
    - *e.g., suppose you have specific restrictions/priors on preferences*
  - but can't handcraft mechanisms for specific settings: hard work!
  - what if we could create one-off mechanisms automatically?

AMD: Basic Setup

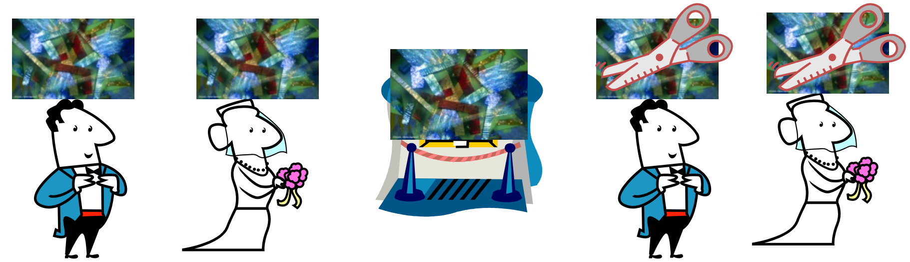
- Assume usual MD setup
  - finite set of outcomes O, *finite* set of (joint) types Θ (restrictive), prior *Pr* over joint types, utility functions
- A direct (randomized) mechanism specified by parameters
  - *probability of outcome* given report: $p(\theta,o)$ for all $o \in O, \theta \in \Theta$
  - *payment* (or transfer to) agent k: $\pi_k(\theta)$ for all k, $\theta \in \Theta$
- Given a social choice *objective* (rather than SCF), optimize choice of these parameters by setting up as a math program (LP or MIP)
  - flexibility in objective (max social welfare, revenue, fairness, minimize transfers, etc…)
- Only complication: need to ensure that parameters are set so that appropriate *incentive* and *participation constraints* are satisfied
  - these can be expressed as linear constraints on the parameters

MIP/LP Formulation

- Objective (example, expected social welfare):
  - $\Sigma_{\theta 1, \ldots, \theta n}\ Pr(\theta_1, \ldots, \theta_n)\ \Sigma_i\ (\Sigma_o p(o \mid \theta_1, \ldots, \theta_n) u_i(\theta_i, o) + \pi_i(\theta_1, \ldots, \theta_n))$
  - many other objectives can be formulated
- Incentive compatibility constraints (example, dominant strategy):
  - $\Sigma_o\ p(o \mid \theta_1, \ldots, \theta_n)\ u_k(o, \theta_k) + \pi_k(\theta_1, \ldots, \theta_n) \geq$
    $\Sigma_o\ p(o \mid \theta_1, \ldots, \theta_k', \ldots, \theta_n)\ u_k(o, \theta_k) + \pi_k(\theta_1, \ldots, \theta_k', \ldots, \theta_n);\ \ \forall k, \theta_{-k}, \theta_k, \theta_k'$
  - Bayes-Nash implementation formulated by taking expectation over θ_{-k}
- Individual rationality constraints (example, ex post IR):
  - $\Sigma_o p(o \mid \theta_1, \ldots, \theta_n)\ u_k(o, \theta_k) + \pi_k(\theta_1, \ldots, \theta_n) \geq 0;\ \ \forall k, \theta$
  - ex interim IR formulated by taking expectation over θ_{-k}
- For randomized mechanisms, this is an LP (assuming linear objective)
  - solvable in polytime (though size proportional to $|\theta||O|$)
- For deterministic mechanisms, this is a MIP (assuming linear objective)
  - even for restricted cases, problem is NP-hard

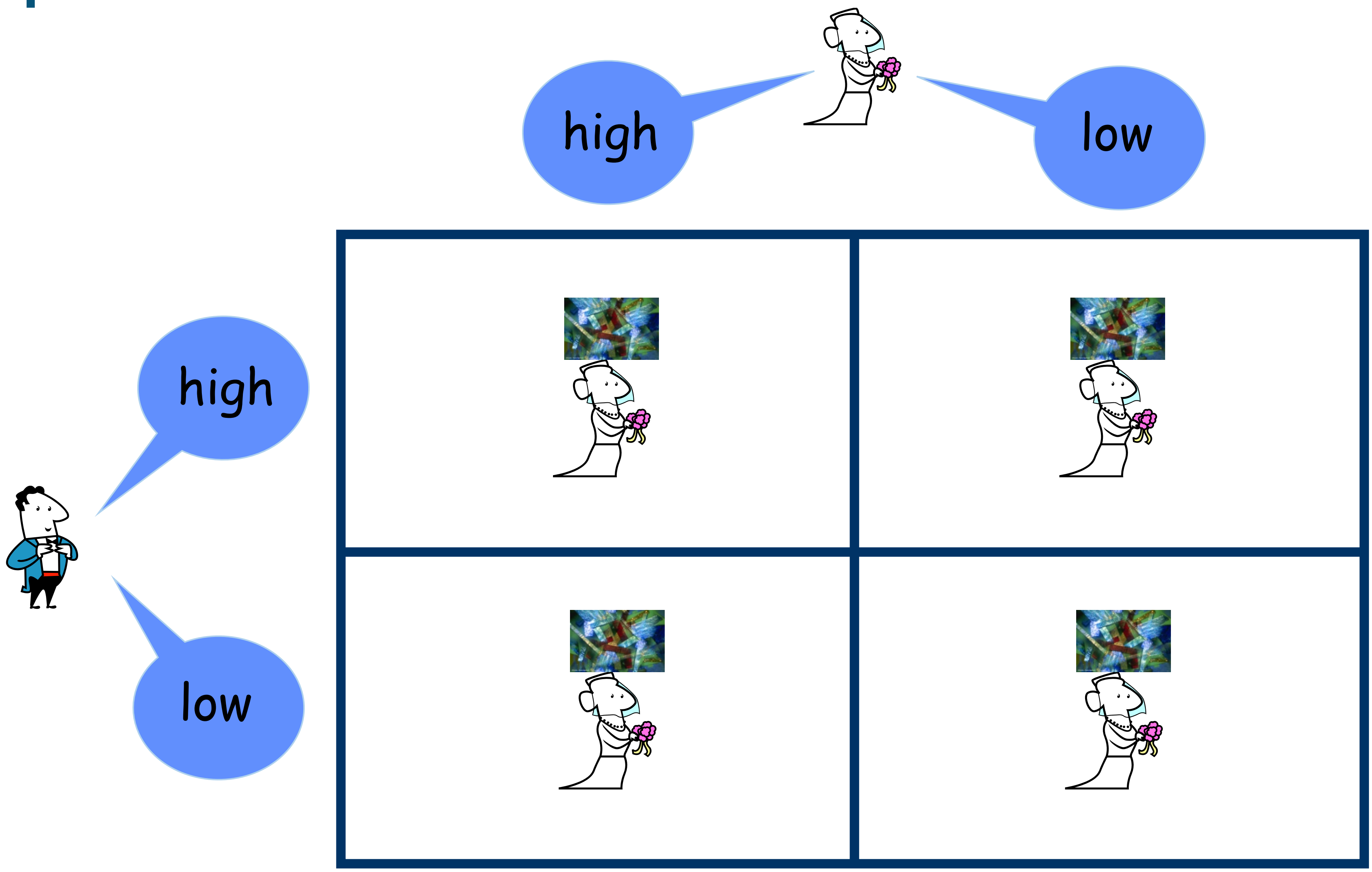
Divorce Arbitration (Conitzer, Sandholm)

- Painting: who gets it
  - five possible outcomes:

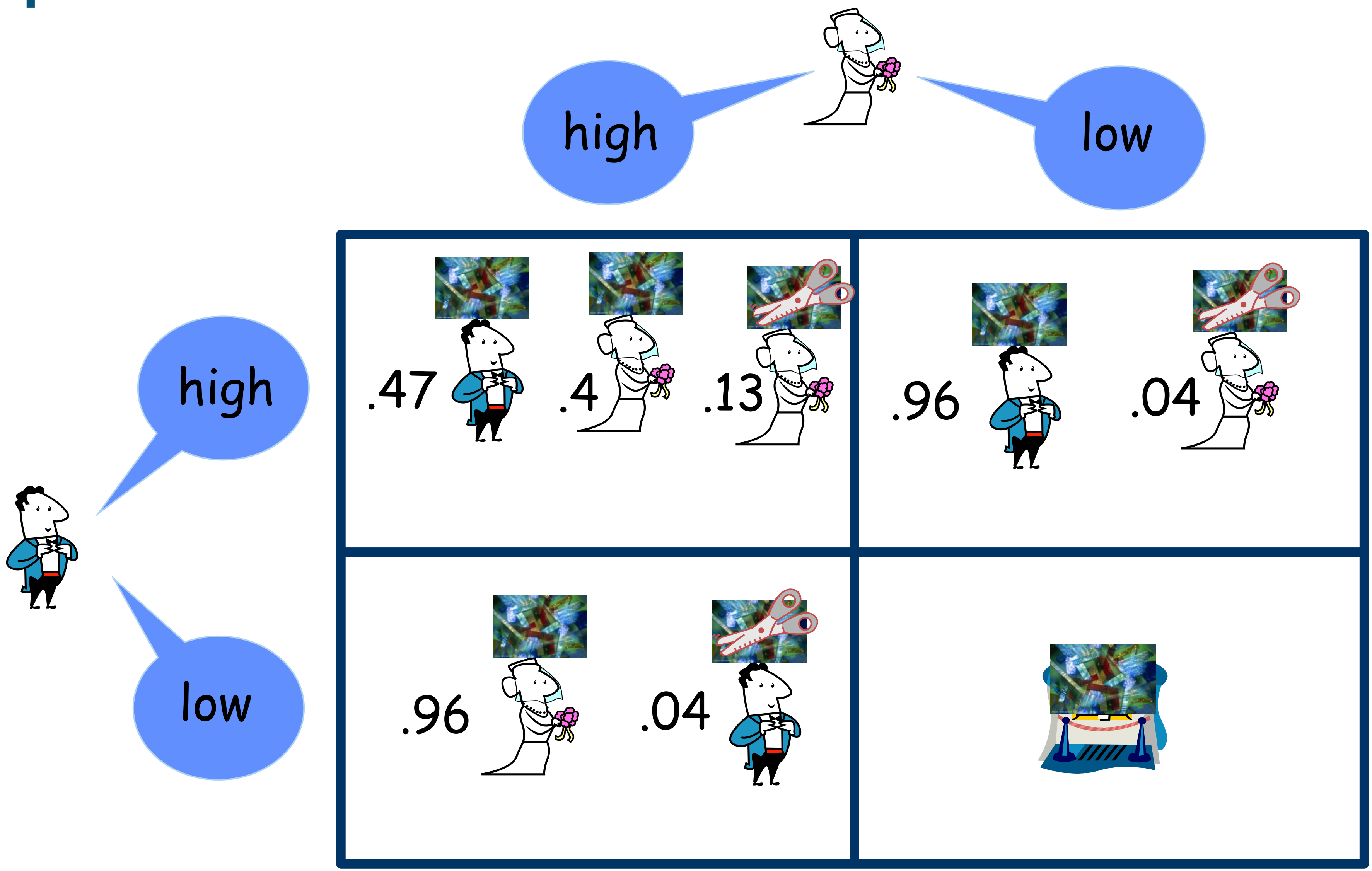


- Two types for husband/wife: high (Pr=0.8), low (Pr=0.2)
- Preferences of *high* type (art lover):
  - u(get the painting) = 110
  - u(other gets the painting) = 10
  - u(museum) = 50
  - u(get the pieces) = 1
  - u(other gets the pieces) = 0
- Preferences of *low* type (art hater):
  - u(get the painting) = 12
  - u(other gets the painting) = 10
  - u(museum) = 11.5
  - u(get the pieces) = 1
  - u(other gets the pieces) = 0

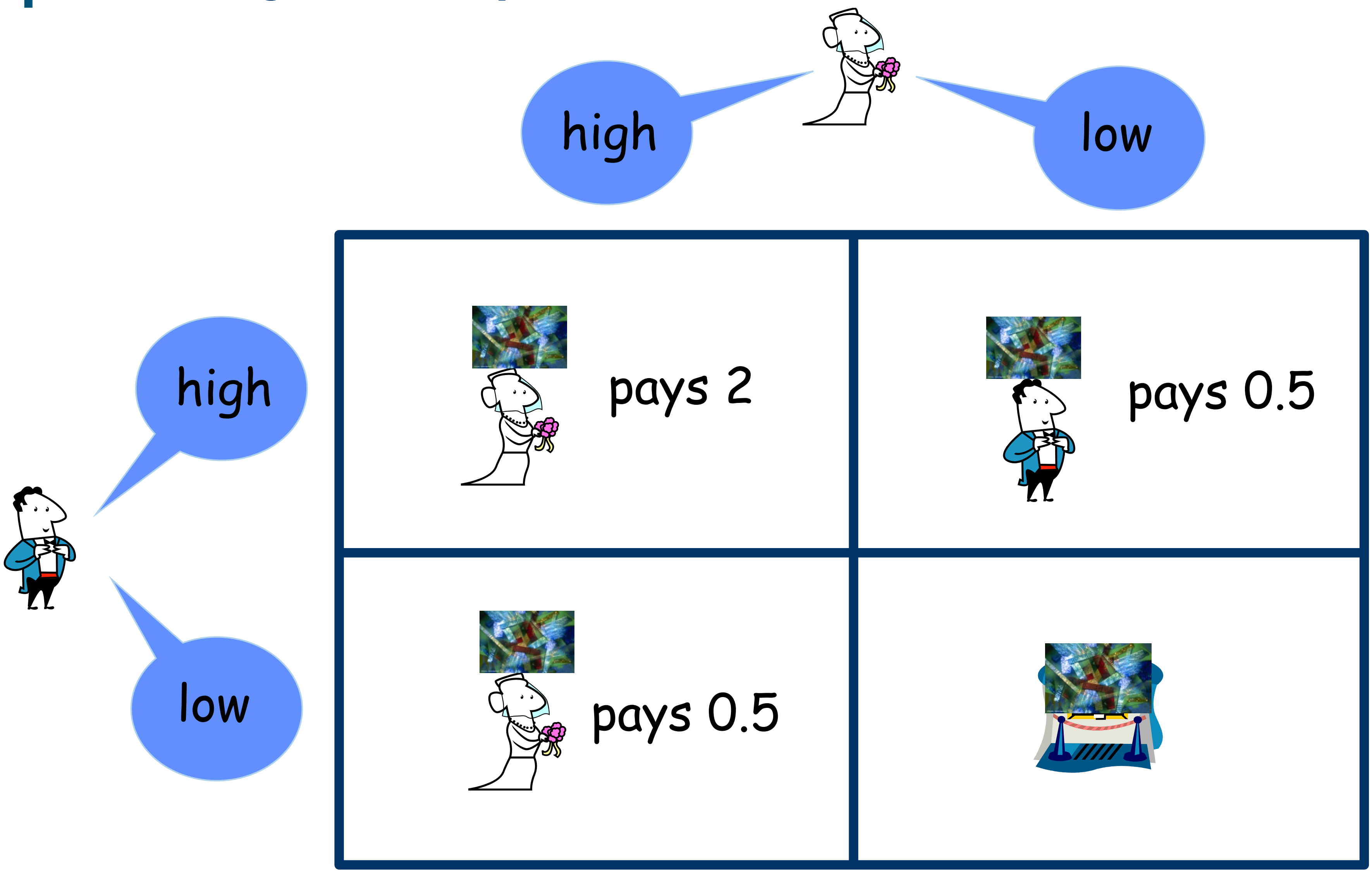
Max Social Welfare (deterministic, no payments)



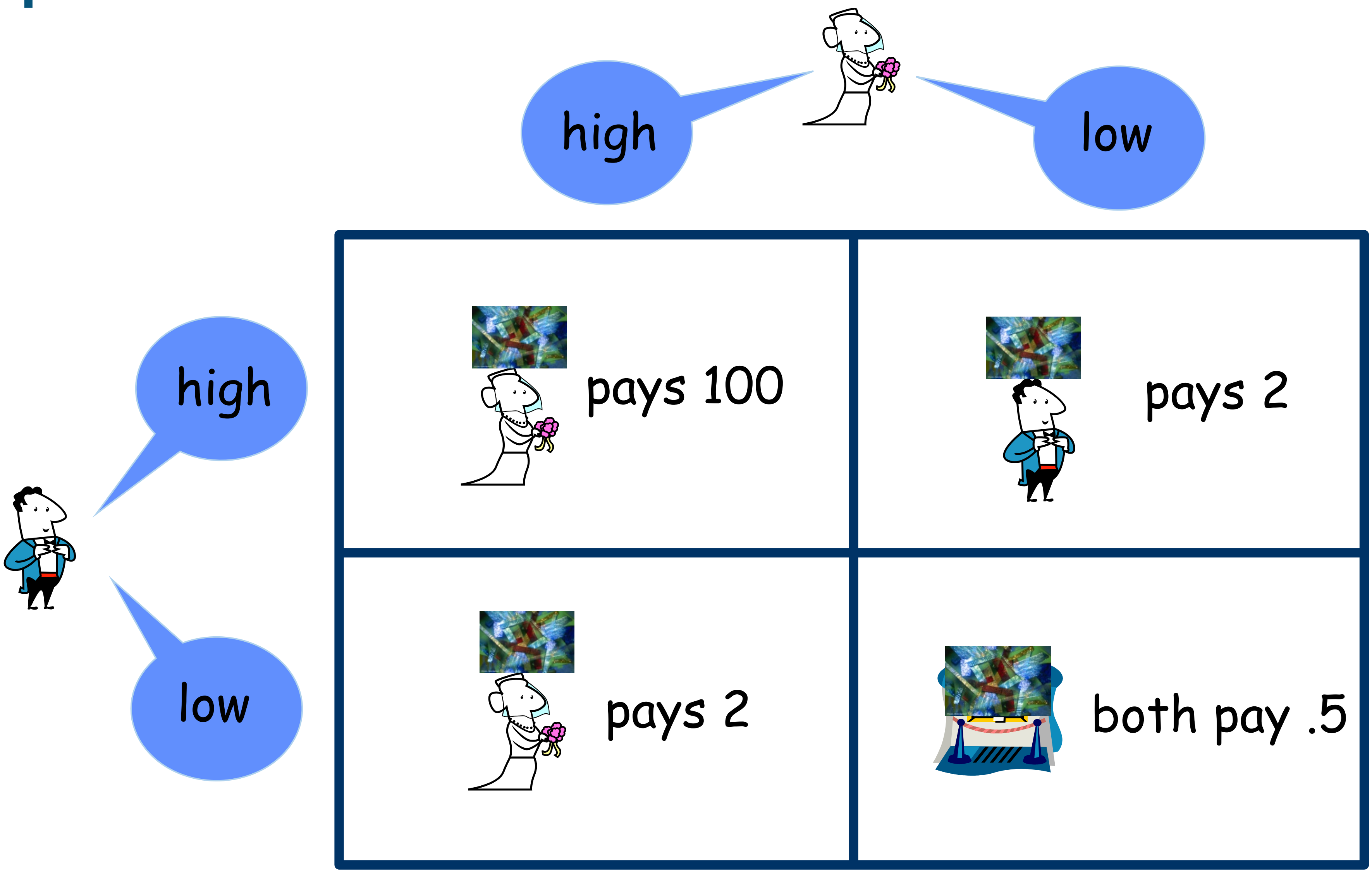
Max Social Welfare (randomized, no payments)



Max Social Welfare (randomized, including payments, excluding "center")



VCG (max social welfare ignoring payments)



AMD: Discussion/Issues to Consider

- Is use of priors in this way acceptable? useful in practice?
- Direct mechanisms:
  - can we avoid full type revelation (especially for large combinatorial spaces, but even just relaxing precision required)
- Related: assumption of finite type space
  - relax by discretization… how best to do this?
  - finite outcome space less problematic (payments broken out)
- Sequential (multi-stage) mechanisms

Partial Type Revelation

- Direct mechanisms assume that preference (type) specification is not a problem for agents
    - but as we saw earlier in course, preference elicitation very hard
- Some work addresses this by allowing agents to specify their valuations/types only partially or incrementally
        - *incremental auctions (English/Japanese, Dutch, CA versions)*
    - **Blumrosen, Nisan, Segal (communication constraints)**
    - Grigorieva et al. (bisection auction)
    - Hyafil and Boutilier (partial revelation VCG)
    - Feigenbaum, Jaagard, Schapira; Sui and Boutilier (privacy)

Limited Communication Auctions

- BNS: limit number of bits bidders use to bid in an auction
  - instead of arbitrary precision, *k* messages (*log(k)* bits)
  - what is the best protocol for *n* agents, each with *k* messages?
    - e.g., maximize (expected) social welfare, or revenue?
- Basic design parameters: choose winner, payments for each tuple of messages received (bid profile)
- Approach: begins abstractly, but proves that optimal auctions have a fairly *natural structure* (we'll work directly with that structure)

- Let's focus on two bidders, social welfare
- Optimal strategies: intuitively, bids correspond to intervals of valuation space, so you can view these as auctions with *"limited precision" bids*

Two-Bit, Two-Bidder Auction: Example

Bidder A \ Bidder B	0	1/4	1/2	3/4
0	B, 0	B, 0	B, 0	B, 0
1/4	A, 1/4	B, 1/4	B, 1/4	B, 1/4
1/2	A, 1/4	A, 1/2	B, 1/2	B, 1/2
3/4	A, 1/4	A, 1/2	A, 3/4	B, 3/4

*each cell shows [winner, price paid]

- Ask each bidder: "Is your valuation at least 0, ¼, ½, ¾?"
  - Threshold strategies (BNS): but we pick thresholds by setting the prices
  - We divide valuation space into intervals: [0, ¼), [¼, ½), [½, ¾), [¾,1]
- Winner: A if bid is "higher" than B; B if higher or tied
  - B has "priority" over A (priority game in the terminology of BNS)
- Payment: minimum bid needed to still win (lower bound of interval)
- Obviously incentive compatible (in dominant strategies)
- Can't guarantee maximization of social welfare
  - if A, B tied, B wins; but A might have higher val (e.g., A: 7/16, B: 6/16)

Two-Bit, Two-Bidder Auction: Different Example

Bidder A \ Bidder B	0	2/7	4/7	6/7
0	B, 0	B, 0	B, 0	B, 0
1/7	A, 1/7	B, 2/7	B, 2/7	B, 2/7
3/7	A, 1/7	A, 3/7	B, 4/7	B, 4/7
5/7	A, 1/7	A, 3/7	A, 5/7	B, 6/7

- Though we don't maximize social welfare, loss can be bounded
  - e.g., if valuations are uniform 0,1, easy to determine expected loss at "ties"
- BNS show that to minimize welfare loss, thresholds should be mutually centered (as in the example above, for uniform [0,1] valuations)
- Also provide analysis of revenue maximization, multiple bidders, etc.

Discussion (Brief)

- Big picture:
  - approach to “partial preference elicitation” in mechanism design
  - derived from a very general “communication” framework
  - trades off communication (cognitive, privacy) for outcome quality
  - BNS are able to obtain DS implementation in SWM case (circumvents Roberts because of restricted valuation space: 1-dimensional)
- Value of partial elicitation more compelling in large outcome spaces (multidimensional)
  - difficulties arise with DS implementation due to Roberts, etc.
  - still there are things that can be done (e.g., by relaxing the equilibrium notions, and bounding incentive to misreport [HB06,07] using minimax regret)